Australian Government
Department of Defence
Defence Science and
Technology Organisation

Challenges and Opportunities in Information Security

Tamas Abraham, David Adie, Angela Billard, Paul Buckland, Samuel Chenoweth, Michael Frangos, Sarath Indrakanti, Martin Lucas, Paul Montague

Command, Control, Communications and Intelligence Division
Defence Science and Technology Organisation

DSTO-TN-1114

ABSTRACT

The biennial Infosec Challenges report provides information to the Defence Signals Directorate (DSD) on a range of current and emerging areas in information security. In our 2012 report, areas have been selected to reflect potential information security interests across a broad range of ICT scenarios in the Australian Government. In each of these areas, we consider the current state-of-the-art, in research and/or practice, and identify existing challenges and opportunities.

RELEASE LIMITATION

Approved for public release

Published by

Command, Control, Communications and Intelligence Division
DSTO Defence Science and Technology Organisation
PO Box 1500
Edinburgh South Australia 5111 Australia

Telephone: (08) 7389 5555
Fax: (08) 7389 6567

AR-015-382
September 2012

APPROVED FOR PUBLIC RELEASE

Challenges and Opportunities in Information Security

Executive Summary

The biennial Infosec Challenges report provides information to the Defence Signals Directorate (DSD) on a range of current and emerging areas in information security. In our 2012 report, areas have been selected to reflect potential information security interests across a broad range of Information and Communications Technology (ICT) scenarios in the Australian Government.

In each area, we have considered the current state-of-the-art, in research and/or practice, and identified existing challenges and opportunities. These areas are:

- The application of Human Computer Interaction (HCI) techniques to computer security, particularly in the area of authentication, by using biometrics, cognitive fingerprints and other contextual interfaces to provide more usable security services.
- Approaches for resilient security that look beyond prevention and detection to incorporate remediation and recovery techniques, enabling ongoing operation in the presence of insecurity.
- Challenges to Service Oriented Architecture (SOA) Security, including the lack of an authorisation standard for SOAs, as well as vulnerabilities affecting SOA-based systems.
- The challenges in implementing and accrediting a Multi-Level Secure (MLS) SOA for Defence, including covert channels, inference and aggregation, and achieving the required levels of certification.
- The risks and challenges yet to be addressed in cloud computing security, including data security, identity and access management (IAM), as well as legal, contractual, governance and policy issues.
- Opportunities to improve data privacy and confidentiality for outsourced computation through expected developments in and application of fully/somewhat homomorphic encryption schemes in the near future.
- Issues that need to be resolved (through policy or research) before personally-owned smartphones and other mobile devices may be integrated into

Government operations in a way that is secure, practical and sensitive to the (sometimes conflicting) needs of the various parties involved.

- Challenges to IPv6 transition, including IPv6 protocol vulnerabilities and flaws in the IPv4 to IPv6 transition mechanisms of dual stack, translation and tunnelling.
- Threats associated with untrusted hardware, the entry vectors and possible damage, as well as the potential for an arms race in attempting to find ways to counteract malicious circuitry.
- The modernisation of critical infrastructure by the introduction of "Smart Grid" systems and the security implications of turning well-controlled, contained systems into a massively distributed network.

Contents

List of Figures

This page is intentionally blank

Glossary of Acronyms

2G	2nd Generation mobile phones
3G	3rd Generation mobile phones (smartphones)
ADL	Architecture Description Language
AES	Advanced Encryption Standard
AGIMO	Australian Government Information Management Office
AMI	Advanced Metering Infrastructure
ANSI	American National Standards Institute
AVDL	Application Vulnerability Description Language
BAN	Business Area Network
BIA	Bump In the Application Programming Interface
BIS	Bump In the Stack
CDE	Cross Domain Environments
CERT	Computer Emergency Response Team
CIS	Customer Information System
CSP	Cloud Service Provider
DAC	Discretionary Access Control
DARPA	Defence Advanced Research Projects Agency
DH	Diffie Hellman key exchange protocol
DNP3	Distributed Network Protocol
DoS	Denial of Service
DSD	Defence Signals Directorate
EUI-64	Extended Unique Identifier, 64 bits long
ETEC	Education Training and Experimental Cloud
ETEN	Education Training and Experimental Networks
FAN	Field Area Network
FHE	Fully Homomorphic Encryption
G3-PLC	3rd Generation Power Line Communications
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications
HAN	Home Area Network
IaaS	Infrastructure as a Service
IAM	Identity and Access Management
IAN	Industrial Area Network
ICMPv6	Internet Control Message Protocol version 6
ICT	Information and Communications Technology
IEEE 802	Institute of Electrical and Electronic Engineers standard number 802
IETF	Internet Engineering Task Force
IHD	In Home Display
IPSec	Internet Protocol Security
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IT	Information Technology
LTE	Long Term Evolution
MAC	Mandatory Access Control

MANET	Mobile Ad hoc Network
MILS	Multiple Independent Levels of Security
MLS	Multi-Level Secure/Security
MSL	Multiple Single Level
NAN	Neighbourhood Area Network
NATO	North Atlantic Treaty Organisation
NETN	NATO Education and Training Network
NFC	Near Field Communication
NIST	National Institute of Standards and Technology
NRL	Naval Research Laboratory
NSA	National Security Agency
NVD	National Vulnerability Database
OFDM	Orthogonal Frequency-Division Multiplexing
OSVDB	Open Source Vulnerability Database
OWASP	Open Web Applications Security Project
PaaS	Platform as a Service
PDA	Personal Digital Assistant
PEMS	Premises Energy Management System
PLC	Programmable Logic Controller
PMI	Privilege Management Infrastructure
QoS	Quality of Service
RBAC	Role Based Access Control
RFC	Request For Comments
RISOS	Research in Secured Operating Systems
ROP	Return Oriented Programming attack method
RSA	Rivest, Shamir and Adleman public key encryption algorithm
RTU	Remote Terminal Unit
SaaS	Software as a Service
SCADA	Supervisory Control And Data Acquisition
SGA	Smart Grid Australia
SIEM	Security Information and Event Management
SIIT	Stateless IP/ICMP Translation
SIM	Subscriber Identity Module
SME	Small and Medium Enterprises
SOA	Service Oriented Architecture
SOX	Sarbanes-Oxley Act
TCP	Transport Control Protocol
TPM	Trusted Platform Module
TRT	Transport Relay Translator
UMTS	Universal Mobile Telecommunications System
USB	Universal Serial Bus
WAN	Wide Area Network
Wi-Fi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
Wireless ISM	Industrial Scientific and Medical radio band
WS	Web Service
WSBPEL	Web Services Business Process Execution Language

WSDL	Web Services Description Language
XML	Extensible Markup Language

This page is intentionally blank

1. Introduction

The biennial Infosec Challenges report provides information to the Defence Signals Directorate (DSD) on a range of current and emerging areas in information security that present:

- challenges to establishing and maintaining information security, and/or
- opportunities to improve information security (including the way it impacts other system objectives – for example, performance, usability etc.) using new techniques.

As the whole-of-government Information and Communications Technology (ICT) security adviser, the application of DSD's policies extend from unclassified or low classification systems, including agency systems whose primary business objectives require a public shopfront interface, to high assurance systems supporting national security and military requirements. The following areas have been selected to reflect potential information security interests across a broad range of ICT scenarios:

- Human Computer Interaction,
- Resilience & security,
- Service Oriented Architecture (SOA) security,
- Multi-Level Secure (MLS) SOA,
- Cloud computing security,
- Fully homomorphic encryption,
- Security risks for mobile device use within the Australian Government,
- IPv6 transition,
- Untrusted hardware, and
- SCADA security.

In each of these areas, we consider the current state-of-the-art, in research and/or practice, and identify existing challenges and opportunities.

2. Human Computer Interaction

Computer systems today usually incorporate various mechanisms to maintain information security, such as normally requiring that human users identify and authenticate themselves before being granted access to the system. Fulfilment of this (and other requirements) depends on having a usable human-computer interaction (HCI) mechanism. For example, identification and authentication are most frequently achieved today through hardware or software keyboard entry of a (user id, password) tuple [1].

There exist commercial platforms that have broken step with some normal security requirements, such as identification and authentication prior to access. For example, Siri on Apple's iOS 5 platform does not (by default) require a user to authenticate himself or herself before providing access to the system [2]. In fact, authentication to smartphones in general is an optional feature that users can configure for themselves. In spite of the increasing sensitivity of smartphones (due to the data they store and process) and, thus, also their viability as a target for malicious activities, allowing users the option to disable these seemingly crucial security features suggests there exists a compelling rationale for doing so. In fact, Smith notes the observations that younger users often perceive security as an obstacle to work around, and that flouting security requirements is seen as a badge of seniority [3].

According to Herley and van Oorschot [1] the main issue for users is usability — one of two key considerations (along with security) in the sub-field of "HCI-Sec", which can be defined as "human computer interaction applied in the area of computer security" [4]. As drivers in the market, many of the HCI trends unfolding today bring with them various HCI-Sec considerations, including unique threats and opportunities. This section will highlight some of these from within the information security framework, focusing first on the traditional area of authentication as a starting point.

2.1 HCI: Authentication

There has been much written about the non-idealities of username-and-password-based security systems, yet their ubiquity still persists to date [5, 6]. Biometric systems and public key infrastructures provide alternatives, but these tend to introduce special requirements, e.g. additional hardware, such as scanners, readers, and user-carried tokens. While many modern systems (such as laptops) often include some kind of biometric hardware (e.g. fingerprint scanners), these alternative architectures typically introduce additional hardware requirements. Combined with factors such as complexity and lack of familiarity (compared to traditional password-based systems), the result is that these alternatives have not been as effective in superseding traditional password-based systems as intended.

Aside from cost and usability, another issue with biometric systems arises from the need to retain or share biometric templates that are used for matching purposes. Organisations and users themselves may object to this on privacy grounds — and rightly so [7] — since biometrics are essentially permanent and irreplaceable. This is unlike passwords or cryptographic keys, which are inherently changeable / revocable and disposable. Similar issues could also apply to emerging software-based biometric paradigms (such as "cognitive fingerprints", see Section 2.1.2), but there is scope to mitigate these security

issues in both traditional and contemporary biometric paradigms (as discussed in Section 2.1.1).

The observed trend towards greater use of software-based techniques within biometric paradigms may prove important for future implementations of transparent multi-factor authentication schemes in pervasive and ubiquitous computing environments. There are three generally accepted authentication factors [8], namely: something the user knows; something the user has; and something the user is. Sometimes a fourth factor is also considered [9]: someone you know, i.e. vouching [10]. Further, there are additional contextual considerations such as location and history that are making inroads into authentication systems [6, 11].

Typically, the more of these factors that are used in conjunction, the more secure the authentication process can be made. On the flip side, this also likely results in greater complexity and cost to implement the system [1], a more onerous process for the user, and, therefore, overall a less usable system that is thus less able to win favour and see widespread adoption or adherence. Consequently, reducing the implementation cost and complexity, while also increasing the usability of multi-factor authentication systems, may be a key research and development goal for realising significant usable security improvements through widespread adoption of these multi-factor systems. To this end, a sample of promising directions will be considered below.

2.1.1 Biometric Security: Cancelability and Cryptosystems

Biometric systems should ideally keep biometric templates secure even in the face of database compromise due to the valid privacy concerns associated with such compromise. Rathgeb and Uhl [12] detail two standard approaches that are being pursued to secure traditional (e.g. finger, iris, face) biometric systems:

1. use of transformation techniques before processing / storing biometric templates — this involves either salting (usually by applying a chosen invertible transformation of) captured biometric data or application of a non-invertible distortion to them; and
2. biometric cryptosystems, which are broadly classified as key binding or key generation schemes. Biometric cryptosystems have generally experienced performance issues that make them too inefficient to consider using.

While current research towards biometric cancelability and cryptosystems has focused on traditional hardware-dependent biometrics, newer software-centric biometric paradigms, such as cognitive fingerprints (discussed in Section 2.1.2), have not yet been scrutinised in depth. However, it is conceivable that cancelability and cryptographic principles can also be applied to similar effect on software biometrics.

2.1.2 Cognitive Fingerprints

One recent initiative being pursued by the United States' Defence Advanced Research Projects Agency (DARPA) takes a fairly pragmatic approach to displacing password-based systems. Their Active Authentication program involves the use of what has been termed a "cognitive fingerprint" [13], which specifically constrains the acquisition of biometrics to

software-based approaches. This mitigates the need for (and associated costs of) additional hardware sensors, while simultaneously also allowing for continuous verification.

While the hardware requirements and continuous verification benefits are realised, these systems may still be conceivably weak against unsupervised input, such as in remote (network) access scenarios — as is the case for traditional biometric systems [14]. For example, a compromised cognitive fingerprint (perhaps acquired by specialised surveillance hardware / software) may be useful in circumventing the legitimate user's cognitive fingerprint in real time — using live data input from a malicious user — thus hiding the true cognitive fingerprint of the current (malicious) user. This is a difficult problem to solve, but is exacerbated in the case of software-based biometrics where the hardware layer of defence provided by hardware-based biometric systems is not present.

Two key factors, among seven identified by Jain, Bolle, and Pankanti [15], commonly used to assess the suitability of biometric systems are acceptability and permanence. *Acceptability* deals with how accepting individuals are to having their particular biometric trait used and *permanence* deals with how invariant a biometric trait is over time. While biometric systems based on cognitive fingerprints are likely to be capable of high acceptability due to their unobtrusiveness, it's unclear how well such systems will score in terms of their permanence, though use of cancelability or cryptographic mechanisms may help mitigate the impact of such an issue. This mitigation may, for instance, involve slowly and transparently evolving credentials derived from the cognitive fingerprint in a manner vaguely resembling password rotation.

As with other biometrics, cognitive fingerprints can be used on their own (uni-modally) or combined with other biometrics (multi-modally). Three possible fusion strategies for multi-modal biometrics are discussed by Heyer [16]:

1. Decision level: all uni-modal matching and decisions are made independently (and possibly asynchronously), then combined for a final decision.
2. Score level: uni-modal inputs provide match scores that are combined to make a final decision — the most commonly preferred option in the literature.
3. Input level: uni-modal inputs are combined into a unified vector for matching and decision making — this option is most flexible, but also data and processing intensive.

The performance of cognitive fingerprints in uni- or multi-modal operation is still largely unknown, though some keystroke- and mouse-based systems, including fusion strategies, have been discussed [17, 18]. But in the context of identification and authentication, cognitive fingerprints may yet prove to be an integral part of secure HCI systems of the future. It may even be possible with minimal changes to existing systems to leverage them for use in lieu of passwords within existing password-based systems, accessed either locally or remotely, as a transitional solution (as alluded to earlier through use of cancelability or cryptographic mechanisms).

2.2 HCI: Horizons

One simple way to reduce the number of barriers faced when integrating new security methodologies is to piggyback on general HCI developments and trends, since these are

usually driven by usability and thus naturally tend to gain wide and fairly rapid acceptance and deployment. Therefore, some notable emerging HCI trends that may be immediately useful for security piggybacking are considered here. Research into how to incorporate useful security functionality within these schemes may prove worthwhile.

In the area of biometric systems, the ISO/IEC 24745:2011 [19] standard attempts to ensure the security of future biometric implementations by providing guidance about biometric and system threats, countermeasures, binding requirements to identities, system application models for different scenarios, and privacy protection for individuals. It is yet to be seen how quickly or well vendors will adopt the standard.

Sabzevar and Sousa [11] suggest that ubiquitous and pervasive computing environments provide opportunities for new contextual multi-factor authentication schemes, which have the additional benefit of adapting to typical user behaviours such as temporarily borrowing another user's credentials. This allows for more dynamic reasoning in relation to authorisation and auditing through an essentially probabilistic authentication paradigm that relies on various HCI vectors.

Jakobsson, Chow, and Molina [6] recognise the continuing "mobile revolution" and the critical demand it creates for easy and fast authentication. In light of the problem of phishing, they argue for new multi-factor authentication techniques that take into consideration the restricted user interfaces we have today, such as small on-screen keyboards that make entry of complex sequences cumbersome. They note that researchers and companies are also experimenting with contextual data, such as history and usage patterns, to improve security in the realm of authentication.

Gartner, the information technology firm, presents a number of technologies in their annual Hype Cycle for Emerging Technologies [20] that appear to be excellent candidates for some of the HCI-Sec opportunities presented, such as:

- Context-enriched services (also touched on above) can provide a rich landscape through which fuzzy / dynamic security-related functions and decisions can occur.
- Location-aware applications provide additional contextual (environmental) data.
- Augmented Reality includes real-time environmental surveillance / monitoring, analysis, and augmentation, which provides novel interaction mechanisms. For example, a current Microsoft Research demonstrator uses projectors and Kinects to create a virtual environment [21].
- Intelligent Software Assistants include platforms such as Siri — an example of a natural language voice platform that could be useful in biometric as well as secret or contextual information inputs.
- Near Field Communications (NFC) and Quick Response (QR) codes provide simple, effortless media for token-based security functionality.
- Computer-brain interfaces are a possible vector for interesting developments in the biometrics sphere.

With various HCI opportunities, there is scope for further research into deriving multiple factors of authentication from simple essential HCI — such as, for example, the use of a Kinect to input a secret gesture while simultaneously deriving a cognitive-fingerprint-

based cryptographic key. Hybrid extensions involving multi-modal biometrics, other knowledge factors, and use of ubiquitous platforms (such as smartphones) could also be pursued.

It seems clear that the evolution of HCI-Sec will depend to a large degree upon the evolution of HCI systems themselves. The challenge is to develop methods to utilise these HCI systems to strengthen their security in ways that still preserve their usability. Significant gains stand to be realised through further research and development of such methods.

3. Resilience and Security

In information and communications technology (ICT), resilience is defined as "the ability ... to provide and maintain an acceptable level of service in the face of various faults and challenges to normal operation" [22].

According to Sterbenz et al [22], resilience is a high-level concept that draws together the following related and overlapping disciplines [22-24]:

- **Fault Tolerance** — the ability of a system to tolerate faults without resulting in service failures.
- **Survivability** — the ability of a system to continue to fulfil its mission, in a timely manner, in the presence of correlated failures (such as intelligent attack or natural disaster).
- **Disruption Tolerance** — the ability to tolerate connectivity disruptions between intermediary components while maintaining end-to-end service between users.
- **Traffic Tolerance** — the ability to continue to carry a normal traffic load in the presence of excessive traffic demands, whether legitimate or malicious.
- **Dependability** — the degree to which a system can be relied upon in terms of availability and continuity of service.
- **Security** — the ability of a system to protect itself from unauthorised access or change.
- **Performability** — the performance of the system in relation to its Quality of Service (QoS) specification.
- **Robustness** – "the trustworthiness (quantifiable behaviour) of a system in the face of challenges that change its behaviour" [22].
- **Complexity** — the ways in which a large number of systems (and, at a lower level, system variables) interact, producing emergent behaviour.

Alternative definitions equate resilience with fault tolerance and place it within a taxonomy where dependability is the higher-level concept [25] or extend survivability to include the concept of recovery [23]. While there may be some disagreement as to exactly how the various disciplines and concepts relate, there is general agreement that resilience goes beyond the prevention and detection of faults to incorporate the concepts of tolerance and recovery [22, 25].

In the definitions given above, security is considered to be one of the disciplines contributing to resilience; however, there is also growing recognition of the need to incorporate resilience approaches within the field of security. In particular, traditional system security is not designed to support ongoing, but degraded, operations after compromise [23, 26, 27]. Network security appliances, such as firewalls, function as filters aimed at the prevention and detection of potential attacks but typically do not provide much in the way of a dynamic response capability [23]. Furthermore, they operate on the principle of keeping attackers out of a bounded, secure system — an architectural assumption that is rapidly losing ground both because of changing architectural

requirements [23] and the prevalence and persistence of cyber threats [27]. Strategies for dealing with security compromise need to develop beyond the unsophisticated isolation and sanitisation technique Garfinkel refers to as the "nuke from orbit and reinstall" approach [26]. What is needed is an approach to operating in the presence of insecurity that supports dynamic and adaptive response to the challenges facing system security.

3.1 Resilience Approaches

There are a number of approaches under research for detecting, operating in the presence of and recovering from system security failures. The following subsections discuss some of these approaches and their challenges.

3.1.1 Key Establishment and Device Attestation

Software Attestation for Key Establishment (SAKE) [28] is a protocol that has been proposed for the dynamic establishment of secret keys between neighbours in a sensor network. A number of attributes of the protocol seek to make sensors resilient to node compromise. These are:

- the ability to establish a secret key without relying on pre-shared secrets;
- the ability to prevent malicious code (present in either participating node) from interfering in the key establishment process or learning the new key; and
- the lack of dependence on side channels, secure communication channels, human intervention or additional hardware.

The SAKE protocol uses the Diffie-Hellman (DH) key exchange protocol in combination with the Guy Fawkes protocol [29] to authenticate DH messages and protect against man-in-the-middle attacks. In addition, the protocol makes use of the Indisputable Code Execution (ICE) primitive [30] to ensure that the protocol has executed on each node without interference from malicious code. The ICE primitive was originally developed to be used as part of Secure Code Update By Attestation (SCUBA) [30], a protocol intended to enable the repair of compromised nodes within a sensor network. While these protocols show promise, Seshadri et al [30] admit that the approach assumes the absence of attackers that are more computationally capable than the sensors participating in the network. In addition, Castelluccia et al [31] point out that the ICE primitive is limited to hardware that permits access to the program counter from software and that it is still susceptible to an attack that allows an attacker to change the location of the ICE protocol execution in memory, pass attestation and then return execution to any code of the attacker's choice, in a Return Oriented Programming (ROP) attack. Other approaches for device attestation in wireless sensor networks include the use of hardware-based Trusted Platform Modules (TPM). Previously, the use of TPMs for this purpose have been considered undesirable due to cost, size and energy constraints [30-32]; however, Tan et al argue that such is no longer the case and present their TPM-enabled Remote Attestation Protocol (TRAP) [32], where each sensor is equipped with a TPM, and provide financial and performance metrics to demonstrate the affordability of the mechanism.

3.1.2 Self-healing Systems

A system is considered to be self-healing if it is able to detect when it is not operating correctly and apply corrective measures without human intervention [33]. The self-healing metaphor and many of the resulting strategies were originally drawn from the biological world, though there is some debate about how worthwhile it is to maintain close parallels between the biology of living organisms and the operation of computer systems [34]. Attempts to closely map between the biological immune system and computer security have resulted in much detailed research on immune-style defences, such as the fundamental and non-trivial task of differentiating between self and non-self [35]. Within the field of autonomous systems, however, there is also ample research on self-protection mechanisms that seek to achieve the purpose, rather than emulate the mechanisms, of the biological metaphor.

Self-healing techniques can operate at a variety of levels. Low-level techniques that focus on detecting and remediating faults within component software or at an operating system level include [36]:

- **Failure-oblivious computing** — uses a compiler that inserts checks to handle writes to unallocated memory.
- **Data structure repair** — detects corrupted data structures and repairs them to meet certain pre-specified constraints.
- **Error virtualisation** — ignores or replaces the results of a failed program 'transaction' with manufactured or predicted return values.
- **Checkpoint-based mechanisms** — rolls an application back to a safe program checkpoint in the event of an error to recover execution flow.

All of these techniques face challenges either in terms of the significant overheads they add to the operation of the program, which may be as high as 2300%, or in terms of the inability to guarantee the semantic integrity of their repairs [36]. This is especially the case if the effects of program transactions are cumulative.

Higher-level techniques focus on self-healing at a compositional level — that is, by restructuring the interconnections between components. These include architectural self-healing systems that express architectures and repair strategies in Architecture Description Language (ADL) [37], middleware that adjusts loads between brokers when it detects a flash-crowd or denial-of-service (DOS) attack [38], as well as routing adjustments in ad hoc networks in the presence of selfish or malicious nodes [39].

One of the challenges of self-healing systems is the ability to ensure that adaptations result in desirable rather than undesirable emergent behaviour. In open-adaptive systems the system is free to 'discover' new adaptations while closed-adaptive systems are limited to a set of prescribed adaptations [40]; however, even closed-adaptive systems can be highly complex and may interact with their environment (and, potentially, other adaptive systems) in unexpected ways. This makes them difficult to formally model and verify in order to gain assurance of their behaviour [34].

3.1.3 ResiliNets

ResiliNets is a strategy for the architecture and design of resilient systems that has been developed in association with the Autonomic Network Architecture (ANA) and ResumeNet projects [22]. The ResiliNets strategy emphasises the need for a multi-layered approach to resilience, both in terms of strategic phases of the resilience life-cycle and addressing the impact of resilience across multiple levels of the protocol stack [41].

The ResiliNets resilience life-cycle is presented as three nested layers of strategic phases [22, 41]:

1. **Passive defence** — structural defence mechanisms including trust boundaries, redundancy, diversity and high connectivity.
2. **Active phases** — performed in real time:
    a. **Defend** — the use of active controls, such as firewalls, to defend against challenges and threats to normal operation.
    b. **Detect** — the ability of the system to detect challenges and recognise when defensive mechanisms have failed to prevent a threat.
    c. **Remediate** — the use of adaptations to system operation in the presence of a challenge in order to minimise the impact on service delivery.
    d. **Recover** — the ability of the system to return from a degraded state to normal operations.
3. **Background phases** — conducted online or offline, by automated or human techniques:
    a. **Diagnose** — analyse the root cause of the problem.
    b. **Refine** — identify refinements for improving the future behaviours of the active phases in the presence of the same or similar challenges.

There is a reasonable body of research within the ResiliNets framework, most of which has been conducted since 2008 [42]. For example, work on developing new challenge identification techniques [43] seeks to identify and address gaps in areas such as real time detection under resource constraints, multi-layer and temporal correlation of events, and the evolvability of challenge detection. Other work [41] seeks to explore mechanisms for achieving multi-level resilience across the network stack, including the development of a multi-level state-space metric for understanding and evaluating the resilience of current and future network architectures.

3.2 Application Areas

Resilience, both in terms of general resilience and resilient security, is an attribute that is broadly applicable across all computing systems, platforms and paradigms; however, there are three key application areas in which recent work in resilience has focussed or is expected to increase. Research in mobile ad hoc networks (MANET) and sensor networks regularly use resilience techniques to address the issue of compromised nodes. This includes the use of remote attestation techniques to recover compromised nodes [28], location-based resilient security to allow graceful performance degradation as the density

of compromised nodes increases [44] and the use of self-healing communities to address routing attacks by non-cooperative nodes [39]. Critical infrastructure protection is an area in which resilience research is expected to grow significantly due to the increasing interdependencies between critical infrastructure and the cyber domain, and the potential for failures to have a high socioeconomic, as well as national security, impact [45, 46]. Resilience techniques already under research for critical infrastructure protection include intrusion tolerance and self-healing using the "Crutial Information Switch" design [46]. Finally, another area in which resilience research is expected to increase is cloud computing. In 2011, DARPA published a funding opportunity for research into Mission-oriented Resilient Clouds (MRC) [47]. The call is based on a recent requirement in the US government for agencies to adopt a "cloud-first" policy when developing new IT deployments. In particular, DARPA highlights three new capabilities to be achieved [47]:

- **Collective immunity —** the use of multiple hosts to provide an increased resistance to coordinated attacks that is scalable and offers flexible trade-offs between attack resistance and overhead.
- **Cloud-wide "public health" infrastructure —** the ability of cloud systems to recognise attacks, assess the trustworthiness of its resources and reallocate resources to ensure high-priority tasks are able to access trustworthy resources in order to support the mission.
- **"Manageable and taskable diversity" and "moving-target defence" —** the development of techniques to make all hosts appear different to attackers, while preserving a common management interface, as well as the use of irregular and diverse task allocation to frustrate an attacker's ability to map the system.

Recently, MIT's Computer Science and Artificial Intelligence Laboratory (CSAIL) has gained funding under the MRC program for their Cloud Intrusion Detection and Repair project [48], which seeks to create a self-healing cloud by using observations of normal cloud operation to derive the properties of secure operation. These properties are then used as a basis for detecting anomalous operations, which prompt the system to intervene and modify those operations, while supporting ongoing service for normal operations [49].

3.3 Conclusion

The concept of resilience has been in evidence for some time, especially in the area of fault tolerance [22, 50]; however, more recently, resilience has taken on a broader meaning. In computer security, in particular, resilience presents the need to think beyond prevention and detection to incorporate techniques for scalable remediation and recovery while still supporting ongoing operations. While there is a substantial body of research in the field, focussing on a variety of approaches and application areas, there are still significant challenges facing the research community in the area of resilience and security, particularly in relation to dealing with complexity, trade-offs, and the ability to model and provide guarantees in the face of emergent behaviour. Research in this area is expected to increase with the need to incorporate resilience into the operation of critical infrastructure and the shared computing resources supplied by cloud computing.

4. SOA Security

Service Oriented Architecture (SOA) is an architectural style and its aim is to achieve a loose coupling amongst interacting distributed systems. SOA is focused on allowing enterprises to develop, interconnect and maintain enterprise applications and services efficiently and cost-effectively. SOA seeks to build upon previous software development efforts such as modular programming, code reuse and object-oriented programming. SOA is designed to assist developers in building applications that interoperate and run seamlessly over heterogeneous environments deployed over multiple platforms. SOA is made up of independent services interconnected via messaging. Platform-independent service interfaces are defined to invoke these services. SOA consists of service providers and service consumers. Service providers define what the service looks like and how to invoke it through an implementation-independent service interface. Service consumers use this interface to invoke the service. SOA also provides a discovery mechanism to act as an intermediary. Service providers publish their service interface using the discovery mechanism for consumers to find and invoke the service.

Services may be invoked synchronously or asynchronously. Synchronous services return a response to the invoker of the service after the service has completed processing the request. Such services typically do not take more than a few seconds to respond. On the other hand, asynchronous services do not return any response to the invoker. However, they may send an acknowledgement of the receipt of the message. At a later time, the service may send the status of processing or the response to the invocation using a call-back mechanism or any other suitable mechanism. These asynchronous messages are correlated using coordination protocols. Exception handling and compensation services are defined to handle exceptions that arise out of invoking services.

SOA offers both business and technical benefits. SOA provides enterprises with the agility to respond to rapidly changing market needs and business requirements by quickly building new applications and updating old applications. The notion of service is understood easily by business people. Therefore, Information Technology (IT) staff within an organisation can interact with them more easily in terms of services. Business processes or workflow services can be explicitly defined so that they can be well understood by both business and technical people. Also, applications or workflow services can be easily outsourced to other organisations, because they can be well defined and their interfaces can be readily specified.

Technical benefits include cost saving by increasing the speed of implementation of any application(s) required and reducing the expenditure on integration technologies. Applications can expose their services in a standard way and existing services can easily be reused. Applications can be exposed more easily to diverse clients over Web-based applications, Personal Digital Assistants (PDAs) and mobile phones.

It is important to note that while SOA and Web services are usually thought to be synonymous, technically they are not. Web services technology is an important tool and *one* implementation mechanism of SOA. However, there may be other implementation mechanisms that are more suitable in any given use case. In this report, when we use the term SOA, we implicitly mean 'SOA implemented using the Web Services technology'.

Although SOA provides many benefits, SOA systems are vulnerable to several security threats. Computer systems and distributed systems in particular face several security risks. They are vulnerable to both active and passive attacks. A distributed system is composed of several layers including a fully functional network, nodes in that network running on a piece of hardware. Operating systems and other software such as middleware are deployed on the hardware in turn enabling running of multiple applications. SOA is a type of middleware in a distributed system, and is therefore vulnerable to security risks affecting each of the layers it is composed of and built upon. Security services such as confidentiality, integrity, authentication, access control/ authorisation, non-repudiation and auditing are used to mitigate such security risks.

4.1 SOA Security Challenges

SOA is affected by classical security vulnerabilities affecting hardware, operating systems, and in turn any software built using the operating systems (see Figure 1). SOA is also affected by Web application vulnerabilities as it is built on top of (thus leveraging) the Internet protocols. We also have a new class of vulnerabilities specifically affecting an SOA, which arise due to the nature of SOA design, and new protocols and message formats supporting an SOA.

Classical security vulnerabilities are those that can be exploited without using more recent Web technologies. An example is buffer overflows [51]. Such vulnerabilities are listed and updated in the U.S. National Vulnerability Database (NVD) [52] using a standard. The standard allows for automated vulnerability management, security measurement, and compliance. There are also other sources that list classical vulnerabilities such as The Open Source Vulnerability Database (OSVDB) [53], US-CERT Vulnerability Notes Database [54], MITRE Common Vulnerabilities and Exposure [55], and SecurityFocus [56]. Research studies on classical vulnerabilities include the RISOS study [57] (vulnerabilities in Operation Systems), the classifications by Aslam et al. [58], Krsul [59], Tsipenyuk et al. [60] and the NRL taxonomy [61]. Other sources of vulnerability classification include books written by Thompson et al. [62] and Howard et al. [63]. As SOA systems leverage existing operating system, software and hardware infrastructure, the security vulnerabilities listed in the sources mentioned above are in general applicable to SOA systems. It is a challenge to mitigate such threats.

The Web Application Security Consortium [64] created the Web Security Threat Classification [65] that clarifies and organises Web applications' vulnerabilities, and develops and promotes an industry standard terminology for describing those vulnerabilities. Similarly, the Open Web Applications Security Project (OWASP) [66] maintains and classifies some of the most critical Web application vulnerabilities. The Application Vulnerability Description Language (AVDL) [67], proposed by the OASIS AVDL TC, is a comprehensive language based on XML that can be used to communicate about specific Web application security vulnerabilities, techniques for discovering those vulnerabilities, and finally security measures to mitigate the corresponding threats. As SOA systems leverage and are built on top of Web technologies, vulnerabilities associated with such technologies also affect SOA systems. It is a challenge to mitigate such Web application threats.

SOA systems are affected by several vulnerabilities in related SOA technologies, protocols and standards such as WSDL [68], WSBPEL [69], and SOAP [70]. For instance, a Web service's metadata may be spoofed thus causing the wrong service to be invoked. Cryptography attacks may cause denial of service by sending very large encrypted XML messages. Harmful SOAP attachments may be sent with encrypted malware thus causing an attack at the server side. Vulnerabilities in XML schema and parsing models are also exploited in some attacks. It is a challenge to mitigate against SOA technology threats. Some of these vulnerabilities and related mitigation strategies have been discussed in [71, 72].

4.2 SOA Authorisation Challenges

Several security standards such as WS-Security [73] have been proposed to provide confidentiality, integrity and authentication services. There is currently no standard available for SOA authorisation. However, several authorisation models have been proposed for Web service authorisation [74-83]. The authorisation model proposed by Indrakanti et al. [81-83] provides the features [84, 85] required for a comprehensive authorisation framework for SOA. Some of the SOA authorisation challenges are mentioned here.

4.2.1 Support for Multiple Access Control Models

The security layer for SOA must provide an authorisation service that is able to support a range of access control models. This is necessary because it is not realistic to expect every SOA application to use the same access control model. In fact, where Web services are used to expose the functionality of legacy enterprise applications, it is likely that organisations will prefer to use their currently existing access control models/mechanisms that they have been using, before exposing the legacy applications as Web services. Therefore, we believe that an authorisation architecture must be generic enough to support multiple access control models including the traditional Discretionary Access Control (DAC) [86], the lattice based Bell-LaPadula Mandatory Access Control (MAC) [87-89], the Role Based Access Control (RBAC) [90], and the Capability/Certificate [91] based access control models.

4.2.2 Authorisation Policies

Languages have long been recognised in computing as ideal vehicles for dealing with the expression and the structuring of complex and dynamic relationships. Over recent years, a language-based approach to specifying access control policies have (rightly) gained prominence, which is helpful for not only supporting a range of access control policies but also in separating out the policy representation from policy enforcement. Hence an important design challenge for SOA authorisation is to enable the support for a range of policy languages for specifying authorisation policies. The policy language(s) used may support fine-grained and/or coarse-grained authorisation policies depending on the organisation's requirement.

4.2.3 Authorisation Credentials

A SOA security service must provide support for defining what access control related credentials are required and how to collect them. Some access control mechanisms may *pull* the credentials from the respective authorities and send them to the responsible authorisation components. For example, in the semantic approach [80], the AC Proxy component collects the relevant privilege (attribute) certificates (for the client) from the PMI Client component which in turn requests the appropriate PMI Node for the privilege certificates for the client. Other access control mechanisms may expect the client to collect the credentials from the respective authorities and *push* them to the responsible authorisation components. For example, Agarwal et al. [74] propose a model in which a client itself collects the required authorisation credentials from the relevant authorities and sends the set of credentials collected before invoking a Web service. Hence, we believe it is an important design challenge for a SOA authorisation model to support both the push and pull models of collecting credentials.

4.2.4 Decentralised and Distributed Architecture

The Web's success is due to its decentralised architecture. Therefore it is reasonable to demand that an authorisation model for SOA should embrace the same decentralised nature of the Web. The need for distribution then follows as a logical consequence. As an example, a company typically comprises a hierarchical internal structure. The decentralised approach allows organisations to specify authorisation policies for Web services on an organisational unit level for different components in the Web service hierarchy. A distributed architecture provides many advantages such as fault tolerance and better scalability and outweighs its disadvantages such as more complexity and communication overhead. Therefore, it is an important design challenge to provide for a decentralised and distributed authorisation model for SOA.

4.3 Conclusion

We introduced SOA security challenges in this report. Classical system vulnerabilities, Web application vulnerabilities, and vulnerabilities affecting SOA-specific technologies are a threat to SOA systems. It is a challenge to make sure all these threats are mitigated in an appropriate and timely manner.

Although several SOA security standards have been proposed in the past, no standard for SOA authorisation has been proposed. We discussed some of the challenges faced when designing a SOA authorisation model.

5. MLS SOA

The use of Service Oriented Architectures (SOA) on Defence networks is increasing due to the benefits it offers, such as improved interoperability, allowing new applications to be built using existing components. Defence operates networks with different levels of security and there are many applications that could benefit from combining services that operate at different levels of security. As a result, the desire for SOA applications to work across different levels of security is also increasing. Designing such a Multi-Level Secure (MLS) SOA presents significant challenges.

5.1 MLS SOA Challenges

5.1.1 Moving MLS to the Application or Services Layer

In the currently implemented approaches, MLS is based on network layer controls or physical separation of security domains [92]. Such approaches are static and inflexible when it comes to SOA as they do not allow sharing of data between services from different security domains. Therefore, to fully leverage the benefits of SOA, such as dynamic service composition and data sharing in order to achieve enterprise goals, MLS needs to move upwards to the application layer where the services and their associated data reside.

5.1.2 Support for Two-way Communications

An MLS SOA solution should be able to support bi-directional communication between domains of differing classifications, without allowing classified data to leak onto a lower classified domain. The requirements for an MLS SOA when using Multiple Independent Levels of Security (MILS) are listed below and are discussed in further detail by Luo and Kang [93].

1. Secure cross-domain service publication and discovery;
2. Secure cross-domain authentication and authorisation of users and services; and
3. Secure cross-domain service invocation.

Note that this requires two-way transactions across the domain boundary.

5.1.3 Covert Channels

Unlike traditional MLS systems, SOA by nature is interactive which increases cross security domain communications. This in turn creates more possibilities for covert channels (such as timing, storage, steganography, etc.) during service interactions. However, if cross domain service interactions are minimised, then the covert channel threats posed to SOAs are not significantly greater than those posed to traditional MLS systems [93].

5.1.4 Release of Identity Information

Unlike traditional MLS systems, SOA is loosely coupled and, more likely than not, will involve the release of identity information across security domains. Traditional authentication models tie trust directly to identity of the user or service. However, the user

or service from a high domain invoking a service from a low domain may need to be anonymous [93]. Therefore, new models of authentication are required for MLS SOA where trust is established even when anonymity is preserved.

5.1.5 Inference and Aggregation

It is well understood that inference and aggregation is a problem when MLS is achieved using cross domain guards [94, 95]. The loosely coupled nature of the SOA means that this classic problem is also applicable to SOA. Therefore, suitable mitigation strategies must be applied on the low end of service interactions so that the identity of a service or user from the high domain cannot be inferred. Also the content of the service interaction itself should be protected from inference and aggregation attacks [93].

5.1.6 Legacy Cross Domain Environment versus Today's Cross Domain Environment Requirements

In [96], the NSA discuss the difference between requirements for traditional cross domain environments (see Figure 1) and today's cross domain environments (see Figure 2). This is briefly summarised here.

In traditional cross domain environments (CDE), typically there were point-to-point connections between domains. However, in today's CDE, as can be seen in Figure 2, cross domain enterprise services are linked to multiple domains. Therefore, enterprise services do not just have point-to-point connections anymore.

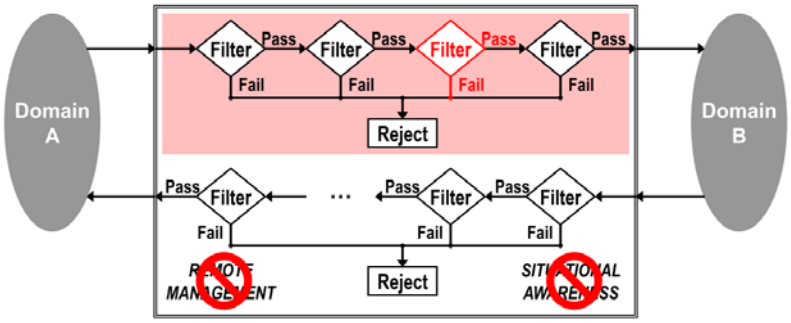

Figure 1: Legacy cross domain environment (adopted from [96])

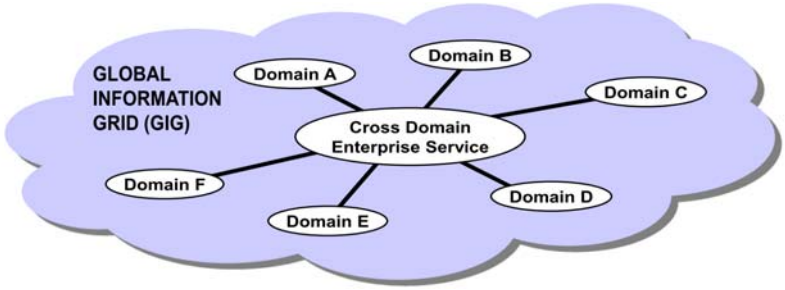

Figure 2: Today's cross domain environment (adopted from [96])

In traditional CDE, there were a limited, static number of data types that needed to be guarded while being transferred between security domains. However, in today's CDE, due to the nature of the services being used, support for a wide variety of data types is required.

Policies for data sanitisation are hard-wired and static in traditional CDE. However, this is not sufficient in today's CDE where support for multiple, dynamic policies is required, again due to the nature of service interactions. This also means remote management of services' policies and the services themselves is required. This was not possible in traditional CDE as there was no situational awareness. However, due to nature of service interactions and the policies associated with them being dynamic, situational awareness is required in today's CDE.

In traditional CDE, any changes to the solution required extensive and expensive (both money and time) certification and accreditation. However, today's CDE should support cost effective and timely certification and accreditation for it to be relevant to the changing landscape of enterprise services.

5.1.7 Accreditation

An MLS SOA solution should be certified and accredited to the level deemed appropriate by the relevant Defence authority. The comprising MLS technologies typically need high levels of certification due to the level of risk involved.

5.2 MLS SOA Solutions

Indrakanti and Buckland [97] surveyed possible MLS SOA solutions for Defence. They identified four distinct architecture patterns in the approaches proposed for achieving MLS SOA. These are listed below.

1. Using Cross Domain Services, e.g. [93, 98, 99],
2. Using Trusted Operating Systems only, e.g. [100, 101],
3. Using cryptography and highly assured components, e.g. [92, 102], and
4. Using Distributed Cross Domain Services for Enterprise Services, e.g. [96].

Using different patterns for different purposes within Defence should also be possible. None of the solutions surveyed have been accredited for use in Australian Defence [97].

5.3 Conclusion

An MLS SOA would provide much benefit to Defence, and a number of solutions have been proposed to implement an MLS SOA. However, there remain significant challenges in satisfying the high level of security required for an MLS SOA in Defence.

6. Cloud Computing Security

Cloud computing is a broad term used to describe a delivery platform for providing IT services over the Internet. As defined by the National Institute of Standards and Technology (NIST)[1], cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (for example, networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. The NIST framework for cloud computing is shown in Figure 3. Cloud computing can deliver benefits to many organisations. In particular, cost savings from the use of cloud computing are attractive to government organisations; however, these benefits come with some additional risks which will be discussed in Section 6.1.

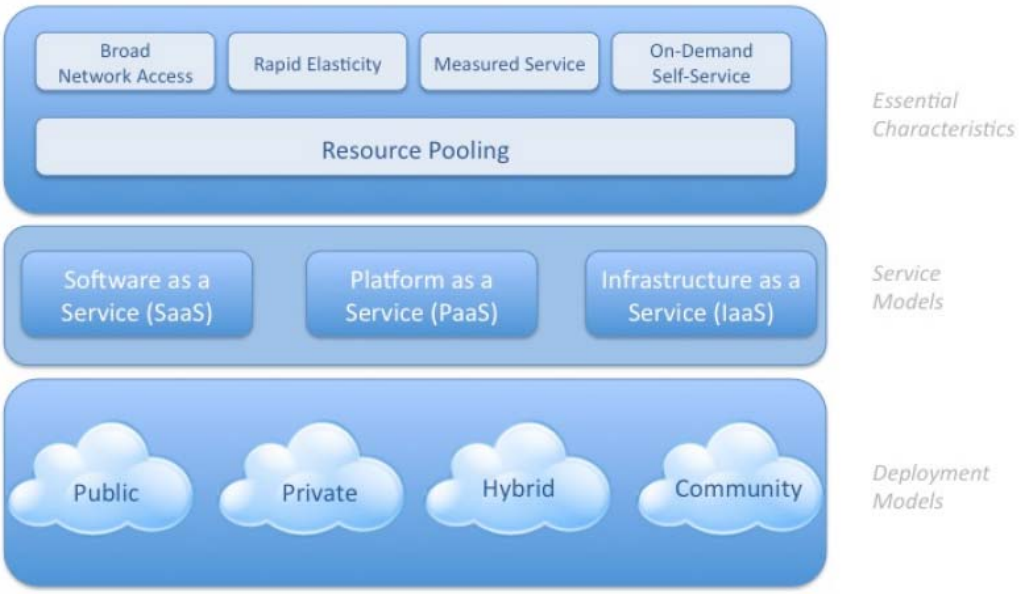

Figure 3: NIST Cloud Computing Framework (adopted from [103]).

There are three types of cloud service models available:

- **Infrastructure as a Service (IaaS)**, where computing resources such as processing power, data storage, network bandwidth, and memory can be leased.
- **Platform as a Service (PaaS)**, where a consumer's software application can be deployed to a cloud.
- **Software as a Service (SaaS)**, where a provider's application is deployed on a cloud and used by consumers over a network.

In addition, there are four cloud deployment models:

[1] National Institute of Standards and Technology, "The NIST Definition of Cloud Computing," document posted October 2009, http://csrc.nist.gov/groups/SNS/cloud-computing/.

- A **Public Cloud** is accessible via the Internet and provisioned by a third party provider on demand to serve consumer needs. The consumer is billed on a fine-grained model for the computing resources utilised.
- **Community Clouds** are used by organisations with similar requirements wanting to share computing resources. Community clouds are cost effective to the individual members. Although the cost involved is higher than leveraging public clouds, there are added benefits in terms of better security, privacy, and policy management and compliance.
- In a **Private Cloud**, the computing resources reside within the organisation's boundary. These resources are either owned or leased. Although the cost involved in buying and maintaining a private cloud is more than other deployment models, it may still be cheaper than maintaining a traditional IT infrastructure. Also, the security of information, the privacy of users, as well as policy management and compliance are under the full control of the organisation.
- A **Hybrid Cloud** is a composition of two or more of the other deployment models. Hybrid clouds are typically used to transition an organisation's IT infrastructure to public clouds whilst complying with security, payment, and other required standards.

6.1 Cloud Security Challenges and Risks

There are a number of security challenges and risks with various types of cloud services and their deployment models. For instance, data security, identity and access management of cloud users, and several legal challenges are faced by both cloud service providers and consumers. We describe some of those challenges in this section.

6.1.1 Data Security

Data stored in the cloud is generally more vulnerable to security threats than data stored in traditional IT infrastructure (for example, data servers). This is because data is co-mingled with that of other customers of the Cloud Service Provider (CSP). The problem becomes worse if the data co-exists with a competitor who may be potentially malicious. In addition, an attack on an enterprise's data means that all other co-located data at the CSP is also potentially vulnerable. Some of the common problems with data security are discussed in [103-107].

Securing customer data at a CSP involves taking care of its confidentiality, integrity and availability. Appropriate encryption and key management techniques must be deployed by a CSP to ensure data is kept confidential. Homomorphic computation may be provided by the CSP to process encrypted input data and produce an encryption of the output (see Section 7). Appropriate data integrity clauses are to be negotiated between a cloud consumer and their CSP. Auditing standards such as the Sarbanes-Oxley Act (SOX2) deal with data lineage and data provenance [106]. Data lineage provides information about where the cloud data came from, where it is stored, and if deleted how it was deleted.

[2] http://www.soxlaw.com/

Data provenance on the other hand proves that data was processed correctly as expected by the cloud consumer. A CSP must appropriately backup data so that it is reliable (available) at all times on demand. On the other hand, when data is deleted on the cloud, the CSP should assure the consumer that strict standards for data remanence [108] are followed. For instance, NIST SP 800-88 [109] provides guidelines for data sanitisation. It deals with data disposal, clearing, purging, and destruction. Security Information and Event Management (SIEM) tools at the CSP should be able to provide auditing and logging information relevant only to a particular consumer by disaggregating such data for legal and forensic purposes.

6.1.2 Identity and Access Management (IAM)

Identity provisioning, authentication, and authorisation are some of the Identity and Access Management (IAM) issues associated with cloud computing [104, 105, 107, 110]. Currently, identity provisioning capabilities offered by CSPs are not standardised and in most cases do not meet enterprise requirements. We recommend both CSPs and enterprises to move towards identity provisioning standards such as Secure Provisioning Markup Language (SPML) [111]. SaaS and PaaS providers typically provide authentication services to individual users, to Small and Medium Enterprises (SMEs), as well as to large enterprises. IaaS providers, on the other hand, usually let the consumers manage authentication to cloud services themselves. CSPs need to control access to their services based on their requirements as well as the consumer requirements. SaaS providers usually specify and enforce their own fine-grained policies for access control of users. On the other hand, consumers of IaaS have a fairly large degree of control over authorisation policies. The IaaS provider only determines coarse-grained policies on the infrastructure. In the case of PaaS, it is a combination of both; the PaaS provider and the consumer share responsibility for specifying and managing their respective authorisation policies based on mutually acceptable criteria.

6.1.3 Legal, Contractual, Governance and Policy Challenges

There are several legal, contractual and policy related issues a cloud consumer needs to be aware of before choosing a CSP. Currently, there is no authority or body responsible for overseeing the way CSPs conduct business. On the one hand, an authority (such as in the financial or other sectors) may help cloud consumers gain confidence that there is always a third party they could involve in resolving disputes with their CSPs. On the other hand cloud computing is a nascent industry and too much regulation may stifle investment (in data centres or other related cloud computing technologies in Australia) or, even worse, discourage innovation. In any case, cloud computing has emerged as a paradigm that is under consideration as a serious alternative to traditional computing, given the cost savings and efficiencies associated with it. Therefore, before new standards and laws for cloud governance emerge, both consumers and CSPs need to work with existing laws or, in some cases, solely based on trust. This means the consumer may start with moving non-critical business services such as instant communication (for example, Windows Messenger) into the cloud. The next step might be to move more critical services into a private cloud where the security and privacy of data is still fully under the consumer's control. The final step is to move core business services into a public cloud once the industry and the related laws and standards have matured. Large enterprises however,

due to the nature of their business and larger budgets, may not move their core services to public clouds in the short or even medium term.

In addition, cloud computing may create significant software licensing issues for enterprises. Clouds in their various forms give a consumer several choices in terms of deployment of software and services. For instance, a consumer wishing to install licensed software on IaaS and PaaS clouds may be violating the terms and conditions of the license. In some cases, it may not be clear what is allowed as the software vendors simply have not kept up-to-date with such use cases (cloud deployments) in the license terms and conditions.

6.1.4 MSL and MLS for Clouds

Cayirci et al. [112] discuss the applicability of cloud computing to military and civil Education Training and Experimental Networks (ETEN). The cloud version of the ETEN is called Education Training and Experimental Cloud (ETEC). Cayirci et al. discuss several security challenges for cloud computing, particularly in the area of Multi-Level Security (MLS). This is because ETEC is used by military applications. Before an MLS cloud can be achieved, they believe multiple single level security (MSL) needs to be achieved in clouds.

North Atlantic Treaty Organisation (NATO) built an ETEN called NATO Education and Training Network (NETN). NETN allows for MSL in its current version by means of security enclaves. Each security enclave has data classified at only one security level. Only users possessing a security clearance equal to or higher than the security enclave's classification can access the information in that enclave. Each enclave sits on a physically separate cloud with its own hardware and software, therefore achieving MSL. Enclaves with different security classification levels can be connected by using Cross Domain Services (CDS) such as data diodes.

The full potential of the NETN can only be achieved, as Cayirci et al. identify, if an MLS solution is realised. In an MLS NETN cloud, users possessing different security clearances should be able to securely access information classified at multiple security levels. As far as we are aware, there is no prototype or commercial MLS solution for cloud computing. Cayirci et al. [112] identified four challenges to be overcome in order to achieve an MLS cloud.

1. Information flow between security enclaves should be reliable. Cayirci et al. believe it can be achieved by labelling data and services with appropriate classifications.
2. Labelling services and data appropriately in a large cloud is a major challenge because of the sheer number of users and the huge size of the databases.
3. A large cloud such as the NETN is used by a large number of users whose clearance requirements may be more dynamic than in traditional systems. To manage such dynamic clearance requirements for a large number of users in real time is a complex task.
4. Efficient sanitisation of documents is the hardest problem to solve. This is because sanitisation techniques must allow a document classified at a higher level than a user's clearance to be viewed by automatically/intelligently removing the parts that cannot be viewed by the user.

6.2 Conclusion

Cloud computing can deliver benefits to many organisations. In particular, cost savings from the use of cloud computing are attractive to government organisations. These benefits come with several security and legal risks and challenges. These risks and challenges can be managed through investment in related research and development activities.

7. Fully Homomorphic Encryption

Homomorphic encryption refers to the property of a cryptographic system to allow computation of an operation (such as addition or multiplication) on data which is available to the evaluating function only as ciphertext. The data resulting from the computation remains encrypted and is the same result that would have been produced had the data had been decrypted, operated on and the result encrypted. In other words, homomorphic encryption allows for the evaluation of functions using sensitive data on an untrusted platform. The utility with regard to cloud computing is immediately apparent. However, an encryption scheme is only required to support a single operation (either addition or multiplication) in order to be considered homomorphic and, as a result, the range of application of such schemes is limited.

On the other hand, in a *fully homomorphic encryption* (FHE) scheme, **both** addition and multiplication of encrypted data are supported. In principle, this would allow any program or computation to be performed using encrypted data. If such a scheme existed and was practical, as we will discuss below, it would no longer be necessary to trust cloud-based servers performing computations with access to sensitive data. Obviously simple cloud-based storage using encrypted data is trivially possible, even in the absence of any homomorphic scheme. Also, in the general computation case, though out of scope of this discussion, we note that residual trust issues remain with regard to whether the cloud has performed the intended operation on the data. At one extreme, denial of service is possible. In more subtle scenarios, the cloud may perform a calculation other than the intended one. Some comments are made with regard to this situation below.

7.1 Initial Feasibility

Whilst there are several efficient cryptographic schemes which are homomorphic in the simpler sense of supporting only a single operation (addition **or** multiplication), even the question of the theoretical existence of a fully homomorphic scheme was for a long time a matter of speculation. Though the general concept of FHE was first proposed by Rivest in 1978 [113], it was not until 2005, that Boneh, Goh and Nissim published a scheme [114] which allows for a single multiplication preceded and/or followed by an arbitrary number of additions. Whilst there were some protocols, for example for secure auctions, developed based upon this scheme, the restriction to a single multiplication operation was a strong limiting factor to its general utility. However, in 2009, Gentry published seminal work [115, 116] introducing an FHE scheme that allows an arbitrary number of both additions and multiplications. This announcement has triggered, over the last few years, a significant amount of research. The initial work of Gentry, however, is far from a practically useful implementation. In particular, both the size of the ciphertext and the computation time are, for practical applications of a reasonable security level, infeasibly large (see discussion of implementation below). The main import of Gentry's initial work is two-fold: (a) demonstrating that FHE is at least theoretically possible, and (b) seeding the subsequent flood of research aimed at developing a practical scheme.

7.2 Theoretical Development

A number of authors [117-124] have focussed on optimisations, simplifications and variants of Gentry's initial scheme. For example, Smart and Vercauteren [119] developed a scheme requiring a smaller key and resulting in lesser message expansion (i.e. a smaller ciphertext), though this is still not a practical solution (see Section 7.3 for a discussion of Implementation). Similarly, Coron et al [120] reduced the size of the public key required for a given set of security parameters from over 800MB to about 10MB, though again still not approaching a practical solution. We will not discuss technical details here, but focus instead (in Section 7.3) on the implementation aspects.

Recent work [125] has proposed a scheme whereby the cloud may be used to securely compute on data from multiple users, extending the FHE paradigm whereby computation is restricted to a user's own data.

Also, recently, Mitchell et al [126] have proposed an expressive language for programming on encrypted data, with the aim of reducing the amount of specialised knowledge required to write secure cloud applications.

7.3 Implementation

Since Gentry's initial work in 2009, the focus of the discussion on FHE has been with regard to whether functions of practical interest may be computed on encrypted data with sufficient security and in a practical time. We discuss, in this section, research so far on implementations of FHE.

We note that any timings quoted below are only given as illustrative of generic trends. The area has yet to reach a point where detailed performance analysis and comparison across schemes at equivalent levels of security is available.

Smart and Vercauteren [119] were among the first to develop an implementation of (their variant of) Gentry's scheme. Although their scheme can theoretically be made fully homomorphic for sufficiently large security parameter values, at values commensurate with practical computation times they were not able to achieve full homomorphism. Hence, the number of multiplications supported by the scheme is limited (of order one to two for the range of values which they explored). At the largest value for the security parameters they were able to test, timings were approximately four seconds for an encryption and three milliseconds for a decryption (timed on a current high-end workstation class machine). However, key generation was not computable in a feasible time within this parameter regime.

However, Gentry and Halevi [127] improved the key-generation method such that the required time is reduced from the order of days to minutes. They have analysed roughly the security of their scheme with the dimension of the underlying lattices, and their analysis suggests that for such secure schemes the key generation step takes of order two hours whilst the "reCrypt" operation (one of the core mechanisms required in order to render the scheme homomorphic) takes approximately 30 minutes. The public key size is of the order of two gigabytes.

Lauter et al [128] consider an implementation of the scheme of [123]. Their focus is on practical applications and corresponding functions which are required to be evaluated;

specifically functions which require only a few multiplications (with many additions). Hence, rather than consider the fully homomorphic scheme, they limit their consideration to "somewhat homomorphic" schemes — that is, schemes supporting a limited number of multiplications only[3]. This allows for considerable improvements in computational efficiency. For example, the cloud's ongoing computation of simple statistical functions of streaming encrypted data will involve mainly addition of the values with a limited number of multiplication operations. The example quoted in the paper is for the computation of the variance of one hundred 128-bit numbers in approximately six seconds (on a mid-range laptop machine).

In addition, [128] proposes use of homomorphically evaluated AES [129] to optimise the size of communications between the client and the cloud. Gentry et al [130] report an average time of about 10 minutes to compute each AES block (16 bytes). Currently, the ciphertext size remains a problem with Lauter et al's specific scheme.

7.4 Applications

The canonical example for applications of FHE is cloud computing. The outsourcing of data to the cloud carries with it concerns regarding data privacy/confidentiality, and these have, in many cases, been sufficient to limit uptake of and migration to the new technologies. The ability to store data in the cloud and further, via use of FHE, to compute functions on that data in the cloud (even, for example, a search of the database can be recast as a function to be evaluated on that data set [129]) provides a powerful, if currently computationally expensive, way to address such concerns.

As we progress ever more towards a mobile computing paradigm with data seamlessly available via cloud services to all of a user's devices, this ability to retain control of access to that data will be critical in many scenarios, particularly with regard to Defence and corporate applications.

Lauter et al [128], however, note a simple example of users' medical data being streamed to the cloud from various monitoring sensors. The cloud, using FHE, can compute statistical functions (in fact, using a "somewhat homomorphic" scheme, as described in Section 7.3) based on the streaming data and various other encrypted user attributes such as age, gender, etc. The user is able to receive and act as appropriate on the computed values, while the privacy of the user data on the cloud is maintained at all times. Other examples discussed include applications using location information to send data to users (in the case quoted, simple advertising) whilst retaining privacy of the users' locations. The retention of privacy in location-based applications is of major import in addressing many of the concerns in this area.

In addition, Lauter et al [128] consider an application in the financial industry in which not only may the data itself (for example, corporate information) be sensitive, but the function evaluated on that data may be sensitive (for example, a predictive model of the stock market performance). It is possible to upload an encrypted version of the function to the cloud so that, as well as the data, the function is hidden from the host cloud machines. We note that this feature may be used to address, at least to some extent, the concern raised

[3] "Somewhat homomorphic" schemes in fact form the basis for the construction of fully homomorphic schemes in Gentry's work. The core of the idea is a way of bootstrapping the latter from the former.

previously with regard to having to trust the cloud to perform the correct computation. If the exact computation is hidden from the cloud, appropriate validation checks may be built into the output (for example, at the extreme end of the scale, a digital signature) to verify that the correct computation has been performed.

Beyond the obvious relevance of the above to government and Defence applications, one may also consider specific Defence-focussed use cases. For example, Schneier [131] has noted the initial claims made by IBM in regard to Gentry's work as enabling spam filters to identify spam even in encrypted emails. One may, of course, use similar techniques to search for dirty words in encrypted data passing through MLS guards. Though a trivial example in itself, it serves to illustrate the possibility of restricting, to a much greater degree than currently, access to the plaintext version of classified documents and files to those entities with a valid need to know.

Another example application might include computation of risk in the cloud whilst maintaining privacy of the user's behavioural and location data on which the computation is based.

We also note that the recent development of multi-party schemes [125] could open up a number of possible new use case paradigms, allowing secure computation across data from multiple users. For example, operations triggering on possible co-location of multiple users could offer useful functionality to users whilst retaining location privacy from the cloud.

7.5 Conclusion

In only the few years which have passed since the discovery of the feasibility of FHE, significant progress has been, and continues to be, made in both the algorithmic and implementation aspects.

Whilst we note that none of the above fully homomorphic implementations are yet truly practical, there are two points to note:

- Improvements in timings from both theoretical improvements and Moore's Law would be expected to render practical fully homomorphic schemes in the relatively near future (of the order of years).
- In practice, for specific applications, a generic fully homomorphic scheme is not actually required. A "somewhat homomorphic" scheme is often sufficient. Optimisations specific to the scheme and the overarching application would be expected to be able to deliver practical implementations within a much shorter timeframe.

With that said, it would be expected that in the near future there will be practical applications for FHE, particularly in the cloud computing space. This will lead to benefits for data privacy and confidentiality for outsourced computation. Whilst perhaps this is not the dawn of a new paradigm, as early reports on the work might have suggested, the niche applications where data confidentiality is of paramount importance, such as Defence and government in general, would seem to warrant the overhead which use of such schemes requires.

8. Security Risks for Mobile Device Use within the Australian Government

In recent years, mobile devices such as smartphones have become ubiquitous in society. As a result, there has been increasing pressure from employees to be allowed to use their own mobile devices for professional purposes, especially smartphones [132]. This is partly because employees do not like the inconvenience of carrying a professional device as well as a personal one, and partly because workplaces frequently use older models and operating systems that the company's information technology staff have had time to evaluate and develop tools and procedures for supporting. Moreover, traditional favourites of the corporate world, such as Windows Mobile and Blackberry, are falling out of favour with users, who increasingly prefer iPhones or Android phones [133, 134].

The introduction of smartphones to Australian government workplaces introduces many security challenges, due to the known vulnerabilities of these devices. In addition, the growing trend towards allowing employees to use personal devices for handling sensitive government information [135] greatly increases the risks involved, since these devices are harder for information technology staff to manage. It is questionable whether the level of trust in such devices that is implied by current policies [135] is warranted.

8.1 Issues

There are many general issues that must be considered when evaluating the security of smartphones and other mobile devices:

- Information stored on the smartphone must remain confidential in the event that the device is lost, stolen or disposed of at the end of its life. This is typically achieved using file system encryption and / or the remote sanitisation feature available on many devices.
- The phone must authenticate users to limit access to the legitimate owner or custodian. This is typically achieved using login passwords or similar, although alternatives, such as facial recognition, exist.
- The phone and carrier network must authenticate each other in order to avoid man-in-the-middle attacks. This is typically achieved using a challenge-response protocol based on a shared secret, which is stored in the network provider's authentication centre and also in a protected chip in the phone's SIM (Subscriber Identity Module) card [136, 137]. In principle, mutual authentication could be handled at the application level instead. In this case, it would be necessary to store all security sensitive material such as private keys and shared secrets in encrypted form, based on the user's password, so that this data cannot be obtained from a stolen phone.
- Data transmitted between the phone and the network must remain confidential. This is typically achieved using encryption [138].
- It must be possible to detect the unauthorised modification of data in transit. This can be achieved by generating a message authentication code that acts as a digest

of the message data, but which can only be generated or verified by a party with knowledge of a shared secret [138] (typically stored in the phone's SIM card and at the network provider's authentication centre). Alternatively, a message authentication code can be generated from the message data alone, which can then be protected from tampering using a digital signature based on public or private key cryptography.

- A consistent security policy must be maintained across different service provider networks and when using a variety of different protocols, such as UMTS (Universal Mobile Telecommunications System), GSM (Global System for Mobile communications), Wi-Fi (Wireless Fidelity), Bluetooth and NFC (Near Field Communication). In particular, security protocols must not be compromised during handover from one protocol to another [139].
- The user must be mindful of the physical environment in which a mobile device is operated. In particular, users must avoid processing private or sensitive information when there is a chance of eavesdropping or 'shoulder surfing' by unauthorised persons.

In addition to the general security issues described above, there are many security issues whose relevance and importance may vary, depending on how mobile devices are used within an organisation. In particular, whether a mobile device is owned and administered by the organisation or is brought to work by an employee is critical in determining the range of security issues that need to be considered.

8.1.1 Issues under the Traditional Corporate Information Technology Paradigm.

In the traditional corporate information technology paradigm, employees are only allowed to store or process their organisation's data using equipment that is owned and administered by the organisation's information technology department. There are obvious security benefits to doing this, such as the existence of a uniform operating system and application environment (which simplifies the task of information technology administration), the ability for information technology staff to roll out software updates across the fleet and the ability to prevent users from modifying critical security settings or installing potentially dangerous applications [132]. However, if this approach is taken for mobile devices, the following issues are introduced:

- The device may be used for mixed personal and professional purposes. This could have implications for the privacy and integrity of any personal data on the organisation-managed phone, with the organisation liable for unauthorised disclosure or loss of the information. It could also lead to conflicts when the mobile device is disposed of at the end of its lifetime, since the employee may want to retain personal data on the phone (e.g. by transferring it to a personal device) but the organisation may wish to wipe all data on the phone according to security policy. Managing such issues is greatly facilitated if personal and professional data can be kept separate on the device.
- In the event of a phone being lost or stolen, the organisation may wish to exercise the right to remotely wipe the device to reduce the chance of information leakage.

This could lead to conflicts with the employee if he / she expects to recover the phone and does not wish to lose personal information on the device.

- Employees who are used to having administration rights on personal smartphones may expect to share administration rights on a phone owned by the organisation and to exercise such rights by installing new applications, etc. Since this may not be allowed under the organisation's information technology security policies, this could lead to conflicts.
- In some situations, it may be desirable to manage personal use of the smartphone, either by limiting it or keeping track of it for the purposes of billing separation. The device may or may not allow personal use to be logged separately. Since excessive personal use at the expense of the organisation or use in contravention of the organisation's policies constitutes fraud, this could be considered a security issue.

8.1.2 Issues under the Bring-Your-Own-Device Paradigm.

In the bring-your-own-device paradigm, employees are allowed to store or process an organisation's data using equipment that employees personally own and administer. If this approach is taken for mobile devices, the following issues are introduced:

- The mixing of personal and professional data on the device could have implications for the privacy and integrity of the professional data, with the employee responsible for avoiding unauthorised disclosure or loss of the information. The difficulty of managing such a responsibility is greatly increased if personal and professional data are not clearly separated on the phone.
- A bring-your-own-device policy could also lead to conflicts when the mobile device is disposed of at the end of its lifetime or if the employee leaves the organisation, since the employee can reasonably expect to sell or retain the phone as he/she sees fit, as well as keeping a copy of all personal data on it. The organisation, on the other hand, may wish to wipe all professional data on the phone according to security policy, which may be difficult to do without destroying personal data if the personal and professional data are not adequately separated on the phone.
- In the event of the loss or theft of a phone, the organisation may wish to remotely wipe the device to reduce the chance of the organisation's information being leaked. This could lead to conflicts with the employee if he / she expects to recover the phone and does not wish to lose personal information on the device. In the case of a device owned by the employee, the organisation's legal ability to force such actions to be taken is greatly reduced [140].
- Employees can reasonably expect to retain administration rights on their personal smartphones even if they are also used for professional purposes. However, this may expose the organisation's sensitive information to a high level of risk, since the employees may not be aware of the dangers of poor security configuration or installing untrustworthy applications.
- The organisation could demand the right to share administrator privileges with the employee as a condition of allowing professional use of the phone and use this to

select appropriate security settings, however there would be nothing to stop the employee from modifying these settings later. The exceptions to this are the iPhone and other iOS products, since their use of configuration profiles allows settings to be introduced that may only be reversed with the knowledge of an associated passcode (not necessarily the user's login passcode) or by wiping the entire phone [141].

8.1.3 Issues Specific to Government

The primary difference between government and non-government organisations (in this context) is that governments handle information on behalf of all their citizens, whereas most non-government organisations only handle information that they own themselves or which they hold on behalf of a limited number of customers. While both have a duty of care to protect the confidentiality of data that is owned by other parties, the unauthorised disclosure of sensitive government data has the potential for more severe and widespread consequences. In addition, Australia's intelligence sharing agreements with various allies means that unauthorised disclosure of information in Australian government custody could have an even wider impact.

As a result of the above considerations, new information and communication technologies need to go through a formal certification process before they can be authorised to handle sensitive government data. The certification process is expensive and time consuming, which means that there can be a significant delay between when new information technology becomes available for commercial use and when (or if) it becomes available for government use. In many cases, technology is already obsolete by the time it is certified and integrated into government systems and processes [142] and may be near the end of its vendor-supported lifetime. This issue is particularly pressing for smartphones, since they are currently one of the more active areas for information technology research and development.

One issue that is unique to government organisations is the need for handling information at multiple classifications. While private organisations may handle quite different types of sensitive information, it is unusual for them to have a formal classification system that is supported by separate network infrastructure at each classification level. In many corporate environments, it would be sufficient for data on smartphones to be segregated into just two divisions, namely the employee's personal data and the corporation's data. In many government organisations, however, it would be necessary to segregate the organisation's data on a smartphone into two or more different classifications, as well as keeping all of these separate from the employee's personal data. For this reason, many smartphone segregation technologies aimed at the commercial market are inadequate for the needs of government organisations, since only two separated domains are provided.

8.2 Smartphone Vulnerabilities and Threats

Smartphones have known vulnerabilities in the following areas:

- User authentication may be very weak or disabled entirely, depending on the device's configuration. While the phone may be set to require strong authentication, it is often impossible to prevent a user with administrator privileges from reversing this. The exception to this are iOS devices, where configuration

profiles may be used to prevent even administrators from weakening security settings.

- Phone to network authentication is not known to be vulnerable to attack when using recent SIM cards [143]. However, version 1 SIM cards (issued prior to 2001) can be cloned [144], facilitating the impersonation of other users' phones to the network.
- The GSM standard does not provide any means for authenticating the network to the phone, facilitating impersonation attacks from rogue GSM base stations. While the UMTS standard does require such authentication for connection to 3G networks, the need for backwards compatibility with GSM means that 3G (3rd Generation) smartphones are still vulnerable to simple 2G (2nd Generation) network impersonation attacks [137].
- In order to eavesdrop effectively and without detection, it is necessary to extend 2G network impersonation attacks to a full man-in-the-middle attack. This could be done if the attacker also impersonates the phone to the network, although that is generally not possible with modern SIM cards. Alternatively, the man-in-the-middle must rely on forwarding of the authentication challenge and response between the legitimate parties in order to authenticate to the network. If the attacker does this, then he/she has no control or knowledge of the keys used for encryption, since these are derived from the shared secret that is known to the phone's SIM card and the network's authentication centre but not the attacker. This would prevent the attacker from eavesdropping but for the fact that the GSM standard does not protect the integrity of all the messages used to negotiate the choice of encryption algorithm [137]. Hence, a man-in-the-middle can eavesdrop by modifying these messages to request the use of very weak encryption or none at all. The only way to prevent such attacks is to disable GSM on the smartphone or warn the user when inadequate encryption is in use (which some smartphones may do although this is not part of the UMTS standard).
- Some smartphones can connect to other devices using a variety of different protocols, such as Bluetooth, NFC and Wi-Fi. The inappropriate configuration or use of these protocols can introduce security vulnerabilities. Bluetooth in particular has several known vulnerabilities [145], although there are measures that may be taken to mitigate some of these [146].
- It is possible for sensitive information to be leaked through side channels, such as stray electromagnetic emissions from a smartphone's circuitry. For example, it has been shown that a loop of wire held in close proximity to a HTC Evo 4G allows the RSA (Rivest, Shamir and Adleman algorithm) private key used by an Android application to be read. A close range attack was also demonstrated on another HTC device doing AES (Advanced Encryption Standard) encryption, as well as a longer range (10 feet) attack on an iPod Touch running an application that performs elliptic curve cryptography [147, 148].
- Applications installed by users can themselves introduce vulnerabilities if they modify settings in a way that weakens security. Even worse, some applications contain malware that may compromise the smartphone when installed or executed.

This is an issue of particular concern on Android phones, owing to the lack of any formal vetting for applications submitted by third party developers [149]. It is less of a problem on iOS devices, since Apple does vet third party applications for malware [150]. The vetting is not foolproof, however; an iOS application was developed recently that was distributed through Apple's application store, but acted as an agent for downloading and installing malware from a remote server [151].

8.3 Possible Solutions and Areas for Future Research

Many of the issues discussed earlier may be addressed by the use of a smartphone with securely separated enclaves, e.g. the Greenhills Platform for Trusted Mobile Devices [152]. This allows the employee to install and administer the operating system of his / her choice on a personal enclave, with a separate professional enclave running a different operating system that is administered by the organisation. If necessary, multiple professional enclaves may be used for handling different security classifications or compartments. This approach allows the employee to conduct both personal and professional business on the one physical device, whilst keeping personal and professional information separate and allowing administrator privileges to be allocated to the appropriate party on an enclave basis.

Further research could be done to solve the problem of security in an uncontrolled physical environment. For example, a private input / output device could be developed which uses a head-mounted display to provide three dimensional visual output directly to the user's eyes (similar to the Z800 3DVisor [153]). Private input could be achieved by displaying a three dimensional keyboard to the user, with hand-tracking sensors used to record keystrokes. To avoid attacks based on observing the user's hands, the keyboard layout would need to be randomised, ideally regularly enough to avoid frequency analysis attacks, although this would also make the keyboard much more difficult to use.

In a further extension of the private input / output device concept, it is possible for the device to include some basic computational capability, so that it can set up an encrypted virtual private network connection to the organisation's application server and then act as a thin client. In this model, the smartphone that the private input / output device plugs into may be treated as part of the untrusted channel (like the Internet is) and thus does not require any formal certification or administrative controls. As long as the private input / output device is certified to be trusted, this arrangement could be used for accessing sensitive information, in public, using a commercial-off-the-shelf smartphone that is entirely under the control of the employee and may be compromised by malware.

There is an extensive body of research on side channel countermeasures. Countermeasures against electromagnetic eavesdropping include improvements to critical hardware to reduce the production of radiation, the addition of shielding to reduce the level of radiation that can escape from the device [154], the deliberate addition of electromagnetic noise to confuse the signal, and randomised delays or randomised processing order to confuse eavesdroppers [155]. This is still an area of active research, however, and it is difficult to be sure if a chosen countermeasure is sufficient to defeat the eavesdropping capabilities of all potential adversaries. With recent government policy changes increasing

the range of sensitive material that may be accessed from mobile devices, this is an area in which more research needs to be done.

8.4 Conclusion

There are many outstanding issues that need to be resolved before smartphones and other mobile devices may be integrated into government operations in a way that is secure, practical and sensitive to the (sometimes conflicting) needs of the various parties involved. While resolving these issues is largely a matter of policy, there is significant scope for further research and development work that could improve the range of policy options available, potentially allowing better outcomes to be obtained.

9. IPv6 Transition

9.1 Introduction

An important challenge currently facing all Australian government agencies is how to manage the transition from Internet Protocol Version 4 (IPv4) to Internet Protocol Version 6 (IPv6) effectively. In 2009 the Australian Government Information Management Office (AGIMO) mandated that all agencies should have IPv6 ready hardware and software in place by the end of 2011 and have all systems IPv6-enabled by the end of 2012 [156]. Even if this 2012 deadline is met (which appears unlikely) [157], it is still expected that IPv6 systems will not completely replace all IPv4 systems in the short term. In fact, to maintain current levels of service, agencies may need to support IPv4 and IPv6 in parallel for a considerable period of time [158]. The core information security challenges involved in the transition to IPv6 are therefore to use the protocol in the most secure manner possible and to ensure that the associated IPv4 to IPv6 transition and coexistence mechanisms are also as secure as possible.

9.2 IPv6 Security

The very fact that IPv6 is a new protocol creates a number of security issues. Firstly, network administrators are likely to be inexperienced in IPv6 deployment and end users will have had little experience with IPv6 capable hosts. If appropriate measures are not taken during initial deployment there is a risk that networks will become compromised. For example, host operating systems may connect to IPv6 networks without explicit user configuration and IPv6 may be exploited by attackers as a "backdoor protocol" [159]. Secondly, networking vendors also lack experience in the production of IPv6 capable devices and some implementations may lack core security features. Implementers of IPv6 devices must comply with over 50 Request For Comments (RFCs), many of which are ambiguous, and confusion resulting from this could lead to products with vulnerabilities and inconsistencies amongst different implementations. As an example, certain network devices may implement deprecated functionality such as the processing of Type 0 routing headers while others may not. The Type 0 routing header has numerous known vulnerabilities and was therefore deprecated by the Internet Engineering Task Force (IETF) in RFC 5095 [160]. Other deprecated functionality, such as site-local addresses, may also lead to vulnerabilities when implemented by certain IPv6 nodes [161].

Most critically though, IPv6 protocol stacks in hosts and network devices have not been as thoroughly tested or exposed to hackers as their IPv4 counterparts. A number of potential vulnerabilities have already been identified which could allow a range of reconnaissance, spoofing, man-in-the-middle, and denial of service (DoS) attacks. These include:

- ICMPv6 Message-based Spoofing — Spoofing of Internet Control Message Protocol version 6 (ICMPv6)-based router advertisements and neighbour discovery packets has implications for the confidentiality and integrity of data, as well as the availability of service [162].

- Fragmentation Attacks — The presence of multiple extension headers in IPv6 packets can be exploited to create fragmented packets that can bypass network protection devices. Alternatively, excessive use of fragmentation can effect denial of service attacks [163].

- Active Network Scanning — This is facilitated by predictable addressing schemes for statically allocated addresses and by predictable EUI-64 (Extended Unique Identifier, 64 bits long) addresses due to known IEEE 802 (Institute of Electrical and Electronic Engineers standard number 802) compatible network card vendors [164].

- Broadcast Amplification Attack – This is a well known denial of service attack in IPv4 networks in which the attacker sends an echo-request message to the subnet broadcast address using the victim's IP address as the source address, thus causing all devices on the subnet to respond and flood the victim with echo-reply messages. Although IPv6 does not define a broadcast address, multicast group messages can be exploited to achieve the same result. The IPv6 protocol attempts to minimise the likelihood of such attacks by strictly forbidding the sending of ICMPv6 messages in response to messages sent to a multicast group [165, 166]. However, the protocol also allows two exceptions to this rule ("Packet too big" and "Parameter problem" ICMPv6 messages) [167]. Hence, broadcast amplification attacks using these specific multicast messages are still possible [168].

- ICMPv6 Error Message TCP Attack — A spoofed ICMPv6 error message sent to the end point of a Transmission Control Protocol (TCP) connection can be used to trigger TCP's fault recovery function which will effectively reset the TCP connection [162].

In addition, according to Lucena et al. [169], at least 22 different covert channels have been identified in IPv6 which may be exploited by attackers for illegitimate communications. It is expected that many more flaws and vulnerabilities will be uncovered as the IPv6 protocol is deployed, hence research efforts must focus on actively discovering such flaws and vulnerabilities, as well as on finding appropriate countermeasures.

9.3 Transition and Coexistence Mechanisms

The Internet Engineering Task Force (IETF) has developed several mechanisms to enable communication during the transition phase [168]. These include:

- dual stack,
- tunnelling mechanisms, and
- translation mechanisms.

Each of these mechanisms have a number of implications for security that information security researchers should address. These are elaborated in the following subsections.

9.3.1 Dual Stack

Dual stack devices have both IPv4 and IPv6 protocol stacks enabled and use each one as appropriate. Hosts and network devices can thus be subject to attack on both IPv6 and IPv4. In fact, IPv4 to IPv6 transition attack tools have long been available that can spoof

and redirect IPv6 packets and launch DoS attacks [166]. Even in a network consisting of dual stack nodes (or machines with IPv6 enabled) running on an IPv4 network, those nodes are still open to local IPv6 attacks [168].

9.3.2 Translation

Translation mechanisms employ protocol translators which act as intermediaries between IPv4 and IPv6 nodes. Mechanisms developed thus far include:

- the Stateless IP/ICMP Translation (SIIT) protocol [170],
- Bump In the Stack (BIS) [171],
- Bump In the Application Programming Interface (BIA) [172], and
- Transport Relay Translator (TRT) [173].

IPv6-IPv4 translation techniques interfere with security protocols such as IPSec and they also provide a means for spoofing and DoS attacks. Relay translation technologies (such as TRT [173]) introduce automatic tunnelling with third parties and thus offer additional DoS possibilities. Furthermore, IPv6-to-IPv4 translation and relay techniques can defeat active defence traceback efforts by hiding the origin of an attack [166].

9.3.3 Tunnelling

Tunnelling mechanisms allow IPv6 packets to be sent and received over an IPv4 network or IPv4 packets to be sent and received over an IPv6 network. There are a number of known security implications associated with such tunnelling. For example, the IPv4 source address of the encapsulating packet, or the IPv6 source address of the encapsulated packet, can be spoofed thus facilitating injection and reflection attacks [168]. Man-in-the-middle attacks are also made possible when tunnelled traffic with inner IP headers containing the attacker's source address and outer IP headers containing spoofed but valid source addresses are used. The attacker may thus breach a network protection device and receive any returned traffic from the (victim) destination node [163]. Furthermore, an attacker may instigate a reconnaissance attack by tunnelling neighbour discovery packets through network protection devices [163]. Whilst these are all known vulnerabilities which can be mitigated using appropriate filtering policies and other techniques [174], there are potentially many more attack vectors associated with IPv4-IPv6 transition mechanisms which have not yet been uncovered.

9.4 Conclusion

The transition from IPv4 to IPv6 for Australian Government departments will be a slow process in which the two protocols will need to coexist for a considerable period of time. Continued research into IPv6 protocol vulnerabilities and flaws in the IPv4 to IPv6 transition mechanisms of dual stack, translation and tunnelling will be critical to ensuring that Government networks and systems are adequately protected throughout this process and beyond.

10. Untrusted Hardware

Securing information within an organisation is a complex task that involves many elements for managing risk. One of these is making sure that the computing systems deployed within the organisation are trusted to perform their functions without exposing the information they carry to unauthorised parties. Often, when such aspects of computer security are addressed, the focus is mainly on software, access control, and to some extent, the physical protection of hardware. As hardware becomes more complex, however, the possibility of unverified, unknown and potentially malicious circuitry appearing in both specialised and commodity computer hardware grows. There is thus a need to identify and minimise such risks when deploying new computing hardware into trusted organisational networks.

The definition of what constitutes untrusted hardware may vary greatly depending on context. It may be that a device does not perform to its specifications and therefore we cannot be sure that it is doing the job it was designed for. It may be that it lacks the necessary protection mechanisms and is unable to ensure the privacy of the information entrusted to it. Or perhaps the device contains malicious circuitry to allow third parties to spy on its operation, accessing and ex-filtrating content that passes through it. The complexity of formally verifying hardware design and the lack of control over manufacturing, transport and installation makes it virtually impossible to be sure that a device is clear of such unwanted circuitry. Where the loss of sensitive information may incur serious consequences, protection against these prospects must be added where possible. This section discusses the challenges presented to an organisation based on the possibility that some of its deployed hardware may contain malicious circuitry. The work below is influenced by an earlier literature review [175].

10.1 Entry Vectors

The most common way malicious hardware can be introduced into the organisation is through regular hardware acquisition. New devices are purchased, such as personal computers, networking components, storage units and displays, with each containing circuitry that has the potential to be malicious. When the device shows up at the doorstep of the organisation, it has already traversed through a number of steps where this circuitry could have been injected into the product:

- During the creation of the integrated circuit,
- During system assembly, and/or
- During supply/transportation.

The opportunities for interference start at the design and manufacture of integrated circuits [176]. Third party tools and software is often used for circuit design, and offshore fabrication is now the norm rather than the exception, allowing malicious inclusions to be inserted into genuine circuitry. Trusted circuits may be replaced by counterfeit or modified hardware during assembly and transport as trust in the supply chain becomes harder to achieve [177]. With the increased adoption of new technologies, threats through mechanisms such as wireless communications [178] are becoming more realistic, particularly when considering the trend to “bring your own” device to work. Where the

malicious circuitry ends up within the organisation (information transfer endpoints such as personal computers and displays versus intermediary points such as routers and gateways) may play an important role in how susceptible the information becomes to loss. A particularly damaging situation can arise when untrusted hardware is inserted by an insider, for example as a malicious peripheral capable of bypassing security policies and use unintended USB channels to communicate with other devices to capture information [179].

10.2 Activity and Damages

A taxonomy has been developed for malicious circuitry [180]. It describes where such logic can be inserted at the development stage, at what physical location and at which abstraction level. It also discusses trigger mechanisms for activating the circuitry from a dormant stage and the types of activity it may perform. These latter comprise of four different classes:

- changing the functionality,
- leaking information,
- degrading the performance, and
- denial of service.

An alternate taxonomy groups the last two types into a single category referring to them as changes in the specification of the circuitry [181].

Examples of the different activities include:

- adding additional circuitry that facilitates privilege escalation [182] (change the functionality);
- using an unused serial (RS-232) port to advertise a cryptographic key [180] (leak information); and
- introducing calculation errors or decreasing the life of a device by accelerating the aging process of its components [176] (change the specification).

10.3 Mitigation

Sensitive information is most susceptible to loss at endpoints: where it is created and where it is consumed; that is, the places it needs to be made available to users. Encryption, for example, may protect against loss at traversal points, such as routers, even in the presence of malicious hardware if the encryption key is secure. To safeguard from information being exposed to other parties via untrusted hardware, an organisation may employ various detection and prevention techniques. Understanding how malicious circuits operate is the first step in these processes, for example, the different ways they can be activated (by sensors, internal state changes, counters etc.) [180].

Detection approaches can be categorised into destructive (for example, using scanning electron microscopy) and non-destructive ones, with further break-down of the latter [176]. Invasive approaches may add further logic to the circuitry, whilst non-invasive ones can analyse operations at run-time or perform various tests (simulation, built-in self tests and side channel analysis) to detect abnormal behaviour caused by the presence of malicious hardware.

A number of preventative approaches at the design, fabrication and post-fabrication stages have been listed in [175]. It is unlikely that they can fully guarantee success due to limited control over these processes, particularly if manufacturing takes place off-shore — jurisdictional issues alone can greatly impact on any influence that can be exerted by the organisation. Instead, focus may be put on preventing malicious circuits from activating [183]; guarding, separating, monitoring, duplicating and evaluating data and results [183]; or using reconfigurable architectures that allow the organisation to re-program the functions of an integrated circuit to meet its own needs [184].

10.4 Conclusion

To date, we are not aware of any confirmed [185] exploits based on malicious circuitry that have been found in the wild. However, akin to the state of affairs experienced by the software security industry, securing hardware may become an arms race: if we can think about an exploit, somebody else may have also thought of it and be some way towards producing it. We must therefore continue to explore how circuits can be compromised and the ways in which we can counteract their damaging potential.

11. SCADA Security

The Smart Grid Australia body (SGA) defines a Smart Grid as "an electricity network that can intelligently integrate the actions of all users connected to it – generators, consumers and those that do both – in order to efficiently deliver sustainable, economic and secure electricity supplies" [186]. Although the rollout of the smart electricity grid is the most advanced, there is also work being done to implement smart gas [187] and water [188] grids.

The deployment of Smart Grids started globally around ten years ago. In Italy 90% of premises are now "smart" and in Sweden the figure is almost 100% [189]. The use of Smart Grid technology has begun in Australia, with trial programs in most states.

11.1 Smart Grid Architecture

The Smart Grid is actually a combination of a number of networks:

1. The Home Area Network (HAN),
2. The Neighbourhood Area Network (NAN),
3. The utility's Backend Office (Billing and Customer Information System), and
4. The utility's Supervisory Control and Data Acquisition (SCADA) network.

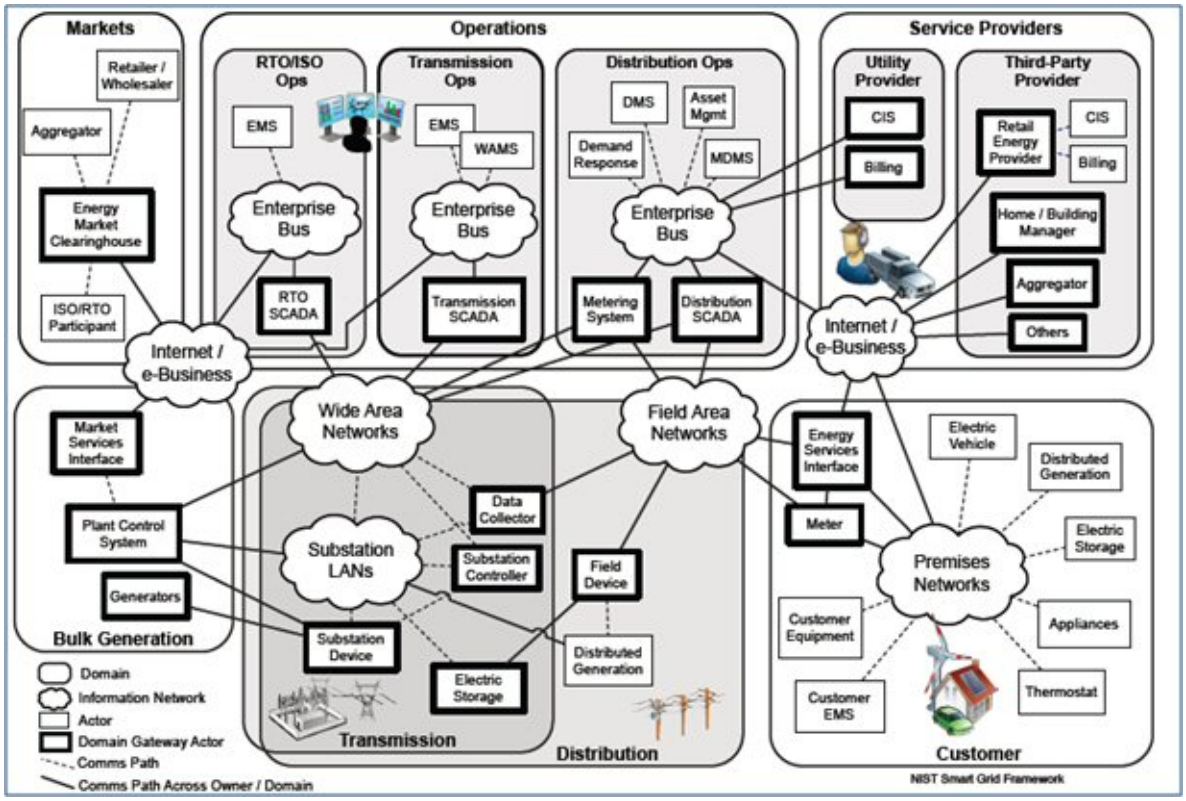

Figure 4: NIST Electricity Grid Conceptual Model, adopted from [190].

Figure 4 shows the architecture of the complete electricity generation and distribution system as defined by the National Institute of Standards and Technology (NIST) [190]. The Smart Grid is the right half of the diagram. In this diagram, the HAN is labelled as "Premises Networks" and NAN is labelled as "Field Area Networks".

11.1.1 HAN

The HAN is the network within the premises, generally anything up to the smart meter, which forms the outer boundary [191]. The equivalent networks in an industrial or business context are an Industrial Area Network (IAN) or a Business Area Network (BAN), respectively. The smart meter is connected to the Premises Energy Management System (PEMS). The PEMS intelligently monitors and adjusts energy usage by interfacing to smart meters, smart devices, appliances and smart plugs. The PEMS provides a level of intelligence implemented in software on dedicated hardware and decoupled from the Smart Meter. The smart meter is intended to have a long lifetime (twenty to thirty years) without any hardware or software changes. In this time the software and firmware on the PEMS would probably have had multiple upgrades.

There are a number of options for the protocol to use for the HAN including [191]:

- Ethernet,
- HomePlug: networking using power line communication,
- Wi-Fi: IEEE 802.11a/b/g/n,
- Z-Wave: proprietary wireless protocol, or
- ZigBee: IEEE 802.15.4.

Ethernet and HomePlug are both wired solutions and the other three are wireless solutions.

The consumer can monitor the energy consumption using an In Home Display (IHD). This may be built into the PEMS or may be located in one or more other locations in the premises. The IHD also allows the consumer to view electricity pricing, usage history and utility messages.

11.1.2 NAN

The NAN provides the link from the neighbourhood smart meters to the nearest access point into the utility's Wide Area Network (WAN). NAN is equivalent to the common terms Advanced Metering Infrastructure (AMI) or Field Area Network (FAN) and is colloquially known as the "last mile". This link could use one of the following communications techniques: Wireless ISM (Industrial Scientific and Medical radio band), IEEE 802.15.4g (proposed new Smart Grid wireless protocol), ZigBee, OFDM (Orthogonal Frequency-Division Multiplexing) power line communication, G3-PLC or broadband power line communication [192]. WiMAX has been used for the last mile in New South Wales for a number of years and trials are underway to use 4G Long Term Evolution (LTE) in the future [193]. Currently long-standing SCADA protocols such as DNP3 and Modbus are commonly used for the last mile but NIST have proposed a new ANSI

standard: C12.22 [194]. The WAN uses long-range and high-bandwidth communication technologies [195].

11.1.3 Utility Provider Back Office Network

This network contains the infrastructure that handles the information and processing capability related to the customers. The Customer Information System (CIS) holds all of the customer account data. The Billing System is responsible for monitoring the usage and billing the customer.

11.1.4 SCADA Network

A SCADA system is a system that monitors and controls devices in an autonomous manner. There are several SCADA systems involved in electricity generation. Figure 4 shows SCADA systems controlling the distribution and transmission of electricity. The generation process is also controlled by a SCADA system, although this is not shown in the figure. The SCADA system consists of a number of servers that connect to controllers known as RTUs (Remote Terminal Units) and PLCs (Programmable Logic Controllers) that control the devices being monitored.

11.2 Security Implications

Traditionally the SCADA systems associated with electricity generation, transmission and distribution, in general, used "security by obscurity". When SCADA systems were first used, they were running on isolated networks, where physical security was the main security issue.

With the advent of the Internet, it was common practice to allow a connection between the corporate network and the control network. There was a good business case for this as it gave management the ability to monitor the status of the system. There may also be other connections into the SCADA system from remote sensors using leased telephone lines, dial-up modem, GPRS or satellite link. It is also possible that there may be connections to a vendor's network. Although these connections are protected by firewalls, it does introduce a possible attack vector into the SCADA system.

With the introduction of the Smart Grid, the SCADA network becomes a huge distributed network, with potentially millions of nodes added to the traditional SCADA network. Additional attack vectors are added due to the possibility that wireless links in the HAN and NAN may be compromised. Wi-Fi, ZigBee and Z-Wave all have ranges of between 20 to 100 metres, allowing an attacker to sniff the traffic and look for vulnerabilities to exploit [191]. It is also possible that access to the network may be obtained by gaining physical access to the hardware, including the smart meters themselves, the PEMS and the base stations in the NAN.

If the NAN is compromised, it may be possible for the attacker to gain access deeper into the system, potentially all the way into the control network. Pollet discusses the security issues facing SCADA systems in general and Smart Grids in particular [196]. He notes that the end devices have limited resources and weak protocol stacks. He makes the comment "Bricking PLCs and RTUs are *(sic)* relatively easy…'Smart Meters' have similar stack issues". An example was given where a Smart Meter was subjected to a Ping Flood, with

varying payload sizes. The results range from the meter recovering after 3 minutes, to the meter having to be rebooted and the configuration reloaded using a serial cable.

Another example was given where the device password, which was stored in flash memory, was overwritten, resulting in the device needing to be sent back to the factory to be reset.

McLaughlin et al. list the following potential attacks on the NAN [197]:

- Physical tampering,
- Password extraction,
- Eavesdropping, and
- Meter spoofing.

Smart Meters used in the US have an optical port that is intended for technicians to be able to diagnose problems with the meter in the field [198]. It is possible for an attacker to connect a laptop to the meter using an optical converter, which can be bought for around US $400. The attacker can then change the settings and data values in the meter. According to Krebs [198] it is possible for the attacker to determine the security code from the device relatively easily. This technique is thought to have been used to defraud an electricity utility in Puerto Rico of $400m in lost revenue, due to meter tampering.

One of the roles of smart meters is to be able to cut off the supply to customers who fall into arrears (known as the "off" switch). They will then be moved from post-paid to pre-paid accounts [199]. Anderson et al. [199] speculate on what might happen if an attacker gained access to the system and sent the "off" command to all of the connected premises. Furthermore, if the attacker sent out a command to change the keys to some thing only known by the attacker or just some random value, the interruption would be made permanent. This would form a massive Denial of Service attack on the affected power grid.

11.3 Privacy Issues

Privacy of the consumer is also an issue, as it is possible to deduce what appliances are being used based on the power usage profile. Ennesser demonstrated this by showing a graph of power usage for a house over the course of a day which displays the use of various appliances at various times [189]. These include a kettle, toaster, washing machine, oven and refrigerator, which each have their own, very distinctive signature.

11.4 Conclusion

The electricity grid is critical to the functioning of a modern society, as are water and gas distribution. These systems are controlled by SCADA systems, which have, in the past, been self-contained systems with minimal and controlled access to outside networks. However there is currently an initiative underway in many countries to modernise the network and push some of the monitoring processing to smart meters at the consumers premises. Although most of the effort has gone into the electricity smart grid, it is likely that the gas and water distribution systems will also be made "smarter".

Some researchers speculate that this new technology is being rolled out without the security implications being fully studied. In effect, the smart grid turns the electricity system from a small well-controlled system, which already has its own set of

vulnerabilities, into a massively distributed network. This greatly increases the potential attack surface for the electricity grid. Some of the communication links in the smart grid use wireless technology. Although the traffic may be encrypted, it allows an attacker to capture network traffic and probe for weaknesses.

It is desirable for further research to be undertaken into the security of the wireless protocols that are proposed for use in the smart grid systems.

12. References

1. Herley, C. and van Oorschot, P., *A Research Agenda Acknowledging the Persistence of Passwords.* Security & Privacy, IEEE, 2012. **10**(1): p. 28-36.
2. Newman, J. *Siri's Security Hole: The Passcode Is the Problem*. PCWorld. 20 October 2011. Available from: http://www.pcworld.com/article/242253/siris_security_hole_the_passcode_is_the_problem.html, [Accessed online: 25 May 2012].
3. Smith, S.W., *Humans in the loop: human-computer interaction and security.* Security & Privacy, IEEE, 2003. **1**(3): p. 75-79.
4. Johnston, J., Eloff, J.H.P., and Labuschagne, L., *Security and human computer interfaces.* Computers & Security, 2003. **22**(8): p. 675-684.
5. Herley, C., van Oorschot, P., and Patrick, A., *Passwords: If We're So Smart, Why Are We Still Using Them?*, in *Financial Cryptography and Data Security*, R. Dingledine and P. Golle, Editors. 2009, Springer Berlin / Heidelberg. p. 230-237.
6. Jakobsson, M., Chow, R., and Molina, J., *Authentication - Are We Doing Well Enough? [Guest Editors' Introduction].* Security & Privacy, IEEE, 2012. **10**(1): p. 19-21.
7. Feng, J. and Jain, A.K., *Fingerprint Reconstruction: From Minutiae to Phase.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2011. **33**(2): p. 209-223.
8. Federal Financial Institutions Examination Council, *Authentication in an Internet Banking Environment*, October 2005.
9. Brainard, J., et al., *Fourth-factor authentication: somebody you know*, in *Proceedings of the 13th ACM conference on Computer and communications security*. 2006, ACM: Alexandria, Virginia, USA. p. 168-178.
10. Yesberg, J.D. and Anderson, M.S., *Quantitative authentication and vouching.* Computers & Security, 1996. **15**(7): p. 633-645.
11. Sabzevar, A.P. and Sousa, J.P., *Authentication, authorisation and auditing for ubiquitous computing: a survey and vision.* International Journal of Space-Based and Situated Computing, 2011. **1**(1): p. 59-67.
12. Rathgeb, C. and Uhl, A., *A survey on biometric cryptosystems and cancelable biometrics.* EURASIP Journal on Information Security, 2011. **2011**(3): p. 1-25.
13. Defense Advanced Research Projects Agency (DARPA), *DARPA-BAA-12-06: Active Authentication - Federal Business Opportunities*. 2012.
14. Balfanz, D., et al., *The Future of Authentication.* Security & Privacy, IEEE, 2012. **10**(1): p. 22-27.
15. Jain, A.K., Bolle, R., and Pankanti, S., eds. *Biometrics: Personal Identification in Networked Society*. The Springer International Series in Engineering and Computer Science. Vol. 479. 1999, Springer. 411.
16. Heyer, R., *Biometrics Technology Review 2008.*, Defence Science and Technology Organisation, DSTO General Document, DSTO-GD-0538, May 2008.
17. Das, R., Mukhopadhyay, S., and Bhattacharya, P., *Continuous Multimodal Biometric Authentication for PC and Handheld Devices.* IETE Journal of Education, 2011. **52**(2): p. 59-69.
18. Tappert, C.C., Villani, M., and Cha, S.-H., *Keystroke Biometric Identification and Authentication on Long-Text Input*, in *Behavioral Biometrics for Human Identification: Intelligent Applications*. 2010, IGI Global. p. 342-367.

19. ISO/IEC, *ISO/IEC 24745:2011, Information technology -- Security techniques -- Biometric Information*. 2011, International Organization for Standardization.
20. Fenn, J. and LeHong, H., *Hype Cycle for Emerging Technologies, 2011* I. Gartner, Research Report, G00215650, 28 July 2011.
21. Lawton, C. *Real-Life Holodecks? Microsoft Kinect Augmented Reality Room Is the Closest Thing Yet*. Wired. 5 November 2011. Available from: http://www.wired.com/geekdad/2011/11/real-life-holodecks-microsoft-kinect-augmented-reality-room-is-the-closest-thing-yet/, [Accessed online: 25 May 2012].
22. Sterbenz, J.P.G., et al., *Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines*. Computer Networks, 2010. **54**(8): p. 1245-1265.
23. Ellison, R.J., et al., *Survivable Network Systems: An Emerging Discipline*. 1997, Software Engineering Institute, Carnegie Mellon University. p. 48.
24. Trivedi, K.S., et al., *Dependability and security models*, in *Design of Reliable Communication Networks, 2009. DRCN 2009. 7th International Workshop on*. 2009. p. 11-20.
25. Avizienis, A., et al., *Basic concepts and taxonomy of dependable and secure computing*. Dependable and Secure Computing, IEEE Transactions on, 2004. **1**(1): p. 11- 33.
26. Garfinkel, S. and Dinolt, G., *Operations with Degraded Security*. Security & Privacy, IEEE, 2011. **9**(6): p. 43-48.
27. Goldman, H., McQuaid, R., and Picciotto, J., *Cyber resilience for mission assurance*, in *Technologies for Homeland Security (HST), 2011 IEEE International Conference on*. 2011. p. 236-241.
28. Seshadri, A., Luk, M., and Perrig, A., *SAKE: Software attestation for key establishment in sensor networks*. Ad Hoc Networks, 2011. **9**(6): p. 1059-1067.
29. Anderson, R., et al., *A new family of authentication protocols*. SIGOPS Oper. Syst. Rev., 1998. **32**(4): p. 9-20.
30. Seshadri, A., et al., *SCUBA: Secure Code Update By Attestation in sensor networks*, in *Proceedings of the 5th ACM workshop on Wireless security*. 2006, ACM: Los Angeles, California. p. 85-94.
31. Castelluccia, C., et al., *On the difficulty of software-based attestation of embedded devices*, in *Proceedings of the 16th ACM conference on Computer and communications security*. 2009, ACM: Chicago, Illinois, USA. p. 400-409.
32. Tan, H., Hu, W., and Jha, S., *A TPM-enabled remote attestation protocol (TRAP) in wireless sensor networks*, in *Proceedings of the 6th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*. 2011, ACM: Miami, Florida, USA. p. 9-16.
33. Ghosh, D., et al., *Self-healing systems -- survey and synthesis*. Decision Support Systems, 2007. **42**(4): p. 2164-2185.
34. Somayaji, A., Locasto, M., and Feyereisl, J., *The future of biologically-inspired security: is there anything left to learn?*, in *Proceedings of the 2007 Workshop on New Security Paradigms*. 2008, ACM: New Hampshire. p. 49-54.
35. Garrett, S.M., *How Do We Evaluate Artificial Immune Systems?* Evolutionary Computation, 2005. **13**(2): p. 145-177.
36. Keromytis, A.D., *Characterizing self-healing software systems*, in *Proceedings of the 4th International Conference on Mathematical Methods, Models and Architectures for Computer Networks Security (MMM-ACNS)*. 2007.

37. Dashofy, E.M., et al., *Towards architecture-based self-healing systems*, in *Proceedings of the first workshop on Self-healing systems*. 2002, ACM: Charleston, South Carolina. p. 21-26.
38. Abie, H., et al., *Self-Healing and Secure Adaptive Messaging Middleware for Business-Critical Systems.* International Journal on Advances in Security, 2010. **3**(1 & 2): p. 34-51.
39. Kong, J., et al., *A secure ad-hoc routing approach using localized self-healing communities*, in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. 2005, ACM: Urbana-Champaign, IL, USA. p. 254-265.
40. Dobson, S., et al., *A survey of autonomic communications.* ACM Trans. Auton. Adapt. Syst., 2006. **1**(2): p. 223-259.
41. Sterbenz, J.P.G., et al., *Redundancy, Diversity, and Connectivity to Achieve Multilevel Network Resilience, Survivability, and Disruption Tolerance (invited paper).* Springer Telecommunication Systems Journal, accepted April 2012.
42. *ResiliNets Publications - ResiliNetsWiki*. Available from: https://wiki.ittc.ku.edu/resilinets/ResiliNets_Publications, [Accessed online: 17 May 2012].
43. Fry, M., et al., *Challenge identification for network resilience*, in *Next Generation Internet (NGI), 2010 6th EURO-NF Conference on*. 2010. p. 1-8.
44. Yang, H., et al., *Toward resilient security in wireless sensor networks*, in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. 2005, ACM: Urbana-Champaign, IL, USA. p. 34-45.
45. Amin, S.M. and Giacomoni, A.M., *Smart Grid, Safe Grid.* Power and Energy Magazine, IEEE, 2012. **10**(1): p. 33-40.
46. Bessani, A.N., et al., *The Crutial Way of Critical Infrastructure Protection.* Security & Privacy, IEEE, 2008. **6**(6): p. 44-51.
47. Defence Advanced Research Projects Agency (DARPA), *DARPA-BAA-11-55: I2O Mission-oriented Resilient Clouds (MRC) - DARPA-BAA-11-55 (Archived) - Federal Business Opportunities: Opportunities*. 2011.
48. Computer Science and Artificial Intelligence Laboratory. *New CSAIL Research Could Help Secure the Cloud | CSAIL*. Massachusetts Institute of Technology. 27 February 2012. Available from: http://www.csail.mit.edu/node/1681, [Accessed online: 17 May 2012].
49. *Center for Resilient Software*. Available from: http://groups.csail.mit.edu/pac/crs, [Accessed online: 17 May 2012].
50. Moore, E.F. and Shannon, C.E., *Reliable circuits using less reliable relays.* Journal of the Franklin Institute, 1956. **262**(3): p. 191-208.
51. Meunier, P., *Software Development and Quality Assurance* in *Handbook of Information Security*, H. Bidgoli, Editor. 2006, John Wiley & Sons Inc.
52. National Institute of Standards and Technology. *National Vulnerability Database.* Sponsored by DHS National Cyber Security Division/US-CERT. Available from: http://nvd.nist.gov/, [Accessed online: 12 January 2011].
53. *The Open Source Vulnerability Database (OSVDB)*. Available from: http://osvdb.org/, [Accessed online: 12 January 2011].
54. US-CERT. *Vulnerability Notes Database*. Available from: http://www.kb.cert.org/vuls/, [Accessed online: 13 January 2011].

55. MITRE. *Common Vulnerabilities and Exposure*. The MITRE Corporation. Available from: http://cve.mitre.org/, [Accessed online: 13 January 2011].
56. Symantec Connect. *SecurityFocus*. Available from: http://www.securityfocus.com/, [Accessed online: 12 January 2011].
57. R. P. Abbott, et al., *Security Analysis and Enhancements of Computer Operating Systems*, Institute for Computer Sciences and Technology, Final Report, NBSIR-76-1041, April 1976.
58. Aslam, T., Krsul, I., and Spafford, E.H., *Use of a taxonomy of security faults*, in *Proceedings of the 19th National Information Systems Security Conference*. 1996, National Institute of Standards and Technology: Baltimore, MD. p. 551-560.
59. Krsul, I.V., *Software vulnerability analysis: PhD dissertation*. 1998, Purdue University.
60. Tsipenyuk, K., Chess, B., and McGraw, G., *Seven pernicious kingdoms: A taxonomy of software security errors.* IEEE Security and Privacy, 2005. **3**(6): p. 81-84.
61. C. Landwehr, et al., *A taxonomy of computer program security flaws.* Computing Surveys, 1994. **3**(26): p. 211-254.
62. Thompson, H.H. and Chase, S.G., *The Software Vulnerability Guide*. Charles River Media Programming Series. 2005: Charles River Media, Inc. 354.
63. Howard, M., LeBlanc, D., and Viega, J., *19 Deadly Sins of Software Security*. 2005: McGraw Hill Osborne Media. 304.
64. Web Application Security Consortium. *Web Application Security Consortium*. Available from: http://www.webappsec.org/, [Accessed online: 13 January 2011].
65. Web Application Security Consortium. *Web Security Threat Classification v2.00*. 1 January 2010. Available from: https://files.pbworks.com/download/p5LJksUNog/webappsec/13247059/WASC-TC-v2_0.pdf, [Accessed online: 13 January 2011].
66. Open Applications Security Community. *The Open Web Application Security Project (OWASP)*. Available from: http://www.owasp.org/index.php/Main_Page, [Accessed online: 13 January 2011].
67. OASIS, *Application Vulnerability Description Language (AVDL) v1.0, http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=avdl*. 2004.
68. World Wide Web Consortium (W3C), *Web Services Description Language (WSDL) v2.0, http://www.w3.org/TR/wsdl*. 2004.
69. OASIS, *Web Services Business Process Execution Language Version 2.0, http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html*. 2007.
70. World Wide Web Consortium (W3C), *SOAP v1.2, http://www.w3.org/TR/soap12-part1/*. 2003.
71. Indrakanti, S., *SOA Security Risks and their Mitigation (to be published)*, DSTO Technical Report, 2012.
72. Jensen, M., et al., *SOA and Web Services: New Technologies, New Standards - New Attacks*, in *Proceedings of the Fifth European Conference on Web Services (ECOWS '07)*. 2007, IEEE Computer Society: Halle (Saale), Germany. p. 35-44.
73. OASIS, *Web Services Security (WS-Security) Specification, http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss*. 2006.
74. Agarwal, S., Sprick, B., and Wortmann, S., *Credential Based Access Control for Semantic Web Services.* American Association for Artificial Intelligence, 2004.
75. Bertino, E., Crampton, J., and Paci, F., *Access Control and Authorization Constraints for WS-BPEL*, in *International Conference on Web Services (ICWS)*. 2006. p. 275-284.

76. Karp, A.H., *Authorization-Based Access Control for the Services Oriented Architecture*, in *Proceedings of the Fourth International Conference on Creating, Connecting and Collaborating through Computing (C5 '06)*. 2006, IEEE Computer Society: Berkeley, CA, USA. p. 160-167.
77. Koshutanski, H. and Massacci, F., *An Access Control System for Business Processes for Web Services*, University of Trento, Technical Report, DIT-02-102, December 2002.
78. Kraft, R., *Designing a Distributed Access Control Processor for Network Services on the Web*, in *ACM Workshop on XML Security*. 2002, ACM: Fairfax, VA, USA. p. 36-52.
79. Mont, M.C., Baldwin, A., and Pato, J., *Secure Hardware-based Distributed Authorisation Underpinning a Web Service Framework*, HPL-2003-144, 17 July 2003.
80. Yagüe, M.I. and Troya, J.M. *A Semantic Approach for Access Control in Web Services*. in *Euroweb 2002 Conference. The Web and the GRID: from e-science to e-business*. 2002. Oxford, UK: British Computer Society, World Wide Web Consortium.
81. Indrakanti, S. and Varadharajan, V., *Coordination based Distributed Authorization for Business Processes in Service Oriented Architectures*, in *The Sixth International Conference on Internet and Web Applications and Services*. 2011: St. Maarten, The Netherlands Antilles. p. 188-194.
82. Indrakanti, S. and Varadharajan, V., *An Authorization Architecture for Web Services*, in *Proceedings of the 19th annual IFIP WG 11.3 working conference on Data and Applications Security*. 2005, Springer-Verlag: Storrs, Connecticut, USA. p. 222-236.
83. Indrakanti, S., Varadharajan, V., and Agarwal, R., *On the design, implementation and application of an authorisation architecture for web services.* International Journal of Information and Computer Security, 2007. **1**(1/2): p. 64-108.
84. Indrakanti, S., Varadharajan, V., and Hitchens, M., *Analysis of Existing Authorization Models and Requirements for Design of Authorization Framework for the Service Oriented Architecture*, in *Proceedings of The 2005 International Symposium on Web Services and Applications (ISWS 2005)*, H.R. Arabnia, Editor. 2005, CSREA Press: Las Vegas, Nevada, USA. p. 35-44.
85. Indrakanti, S., Varadharajan, V., and Hitchens, M., *Principles for the Design of Authorization Framework for the Service Oriented Architecture*, in *International Conference on Internet Technologies and Applications (ITA 05)*. 2005: Wrexham, North Wales, UK.
86. Lampson, B.W., *Protection*, in *5th Princeton Symposium on Information Science and Systems*. 1971. p. 437-443.
87. Bell, D.E. and LaPadula, L.J., *Secure Computer Systems: Unified Exposition and Multics Interpretation*, 1975.
88. Bell, D.E. and LaPadula, L.J., *Secure Computer Systems: Mathematical Foundations and Model*, National Technical Information Service, Mitre Technical Report, M74-244, 1974.
89. Bell, D.E. and LaPadula, L.J., *Secure Computer Systems: Mathematical Foundations*, National Technical Information Service, Mitre Technical Report, ESD-TR-73-278, November 1973.
90. Sandhu, R., et al., *Role-Based Access Control Models.* IEEE Computer, 1996. **29**(2): p. 38-47.
91. Chadwick, D.W. and Otenko, A., *The PERMIS X.509 role based privilege management infrastructure.* Future Generation Computer Systems, 2003. **19**(2): p. 277-289.

92. Waller, A. and Yau, A., *Multilevel Security for SOA: Position Paper*, THALES, 13 October 2009.
93. Luo, J. and Myong Kang, *An Infrastructure for Multi-Level Secure Service-Oriented Architecture (MLS-SOA) Using the Multiple Single-Level Approach*, N.R. Laboratory, Memorandum Report, NRL/MR/5540--09-9220, 17 December 2009.
94. Marks, D.G., *Inference in MLS database systems.* IEEE Transactions on Knowledge and Data Engineering, 1996. **8**(1): p. 46-55.
95. Garvey, T.D. *The inference problem for computer security*. in *Computer Security Foundations Workshop V*. 1992. Franconia, NH.
96. NSA Cross Domain Solution Products Division, *Distributed Service Oriented Architecture (SOA)-Compatible Cross Domain Service (DSCDS) Overview*, in *Unified Cross Domain Management Office Conference 2009*. 2009: San Diego, CA.
97. Indrakanti, S. and Buckland, P., *MLS SOA Architecture Survey*, DSTO Client Report, DSTO-CR-2011-0073, January 2011.
98. Ramasamy, H.V. and Schunter, M., *Multi-Level Security for Service-Oriented Architectures*, in *Military Communications Conference (MILCOM), 2006*. 2006, IEEE Press: Washington, DC. p. 760-766.
99. Raytheon. *High Speed Guard*. Available from: http://www.raytheon.com/capabilities/products/cybersecurity/highspeedguard/index.html, [Accessed online: 28 October 2010].
100. Raney, C.J., *Integrating Multilevel Command and Control into a Service Oriented Architecture to Provide Cross Domain Capability*, S.a.N.W.S.C. (SPAWAR), 2006.
101. Sauer, L., et al., *Towards achieving cross domain information sharing in a SOA-enabled environment using MILS and MLS technologies*, in *Military Communications Conference*. 2009, IEEE: Boston, MA. p. 1-5.
102. Alten, A.I., *Designing a Large SOA across Multiple Security Domains*, in *Military Communications Conference, 2007. MILCOM 2007. IEEE*. 2007. p. 1-8.
103. Cloud Security Alliance (CSA), *Security Guidance for Critical Areas of Focus in Cloud Computing V2.1*, December 2009.
104. Buecker, A., et al., *Cloud Security Guidance: IBM Recommendations for the Implementation of Cloud Security*, IBM Corporation, IBM Redpaper, REDP-4614-00, 2 November 2009.
105. European Network and Information Security Agency (ENISA), *Cloud Computing: Benefits, risks and recommendations for information security*, 2009.
106. Mather, T., Kumaraswamy, S., and Latif, S. *Cloud Security and Privacy*. [O'Reilly Webcast]. 24 September 2009. Available from: http://www.youtube.com/watch?v=tF2EV5olkbQ, [Accessed online: 30 November 2010].
107. Mather, T., Kumaraswamy, S., and Latif, S. *Cloud Security Deepdive*. [O'Reilly Webcast]. 20 January 2009. Available from: http://www.youtube.com/watch?v=Tt09qFeZF0Y, [Accessed online: 30 November 2010].
108. Hughes, G.F., Coughlin, T., and Commins, D.M., *Disposal of Disk and Tape Data by Secure Sanitization.* IEEE Security and Privacy, 2009. **7**(4): p. 29-34.
109. Kissel, R., et al., *Guidelines for Media Sanitization*, National Institute of Standards and Technology, NIST Special Publication, SP 800-88 September 2006.

110. Cloud Security Alliance (CSA), *Domain 12: Guidance for Identity & Access Management V2.1*, April 2010.
111. OASIS, *Service Provisioning Markup Language (SPML) Version 1.0, www.oasis-open.org/committees/download.php/4137/os-pstc-spml-core-1.0.pdf*. 2003.
112. Cayirci, E., et al., *Snow Leopard Cloud: A Multi-national Education Training and Experimentation Cloud and Its Security Challenges* in *Proceedings of the 1st International Conference on Cloud Computing (CloudCom '09)*, M.G. Jaatun, G. Zhao, and C. Rong, Editors. 2009, Springer-Verlag: Beijing, China. p. 57-68.
113. Rivest, R.L., Adleman, L., and Dertouzos, M.L., *On data banks and privacy homomorphisms*, in *Foundations on Secure Computation, Academia Press*. 1978. p. 169-179.
114. Boneh, D., Goh, E.-J., and Nissim, K., *Evaluating 2-DNF Formulas on Ciphertexts*, in *Second Theory of Cryptography Conference, TCC*. 2005. p. 325-341.
115. Gentry, C., *Fully homomorphic encryption using ideal lattices*, in *Proceedings of the 41st annual ACM symposium on Theory of computing*. 2009, ACM: Bethesda, MD, USA. p. 169-178.
116. Gentry, C., *A fully homomorphic encryption scheme*. 2009, Stanford University.
117. van Dijk, M., et al., *Fully Homomorphic Encryption over the Integers*, in *Advances in Cryptology -- EUROCRYPT 2010*, H. Gilbert, Editor. 2010, Springer Berlin / Heidelberg. p. 24-43.
118. Stehlé, D. and Steinfeld, R., *Faster Fully Homomorphic Encryption*. Advances in Cryptology ASIACRYPT 2010, 2010. **6477**: p. 377-394.
119. Smart, N.P. and Vercauteren, F., *Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes*, in *Public Key Cryptography*. 2010. p. 420-443.
120. Coron, J.-S., Naccache, D., and Tibouchi, M., *Public Key Compression and Modulus Switching for Fully Homomorphic Encryption over the Integers*, in *Advances in Cryptology -- EUROCRYPT 2012*, D. Pointcheval and T. Johansson, Editors. 2012, Springer Berlin Heidelberg. p. 446-464.
121. Brakerski, Z., Gentry, C., and Vaikuntanathan, V., *Fully Homomorphic Encryption without Bootstrapping*. 2011.
122. Gentry, C. and Halevi, S., *Fully Homomorphic Encryption without Squashing Using Depth-3 Arithmetic Circuits*. 2011.
123. Brakerski, Z. and Vaikuntanathan, V., *Efficient Fully Homomorphic Encryption from (Standard) LWE*, in *Proceedings of the 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*. 2011, IEEE Computer Society: Washington, DC, USA. p. 97-106.
124. Brakerski, Z., *Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP*. 2012.
125. López-Alt, A., Tromer, E., and Vaikuntanathan, V., *On-the-Fly Multiparty Computation on the Cloud via Multikey Fully Homomorphic Encryption*, in *Proceedings of the 44th symposium on Theory of Computing (STOC '12)*. 2012, ACM: New York, NY, USA. p. 1219-1234.
126. Mitchell, J.C., et al., *Information-flow control for programming on encrypted data*. 2012.
127. Gentry, C. and Halevi, S., *Implementing Gentry's fully-homomorphic encryption scheme*, in *Proceedings of the 30th Annual international conference on Theory and applications of cryptographic techniques: advances in cryptology*. 2011, Springer-Verlag: Tallinn, Estonia. p. 129-148.

128. Lauter, K., Naehrig, M., and Vaikuntanathan, V., *Can Homomorphic Encryption be Practical?* 2011.
129. Smart, N.P. and Vercauteren, F., *Fully Homomorphic SIMD Operations*. 2011.
130. Gentry, C., Halevi, S., and Smart, N.P., *Homomorphic Evaluation of the AES Circuit*. 2012.
131. Schneier, B., *Homomorphic Encryption Breakthrough*, 9 July 2009, Available from: http://www.schneier.com/blog/archives/2009/07/homomorphic_enc.html, [Accessed online: 23 May 2012].
132. Schneier, B. *Should Enterprises Give In to IT Consumerization at the Expense of Security?* , Schneier on Security. Available from: http://www.schneier.com/essay-323.html, [Accessed online: 22 March 2012].
133. Pennington, S. *BlackBerry losing corporate popularity*. The Age. Available from: http://www.theage.com.au/it-pro/business-it/blackberry-losing-corporate-popularity-20111017-1lswi.html, [Accessed online: 22 March 2012].
134. Turney, D. *BlackBerry tries to remain relevant*. The Age. 23 January 2012. Available from: http://www.theage.com.au/it-pro/business-it/blackberry-tries-to-remain-relevant-20120123-1qd6j.html, [Accessed online: 22 March 2012].
135. Defence Signals Directorate. *iOS Hardening Configuration Guide - FOR iPOD TOUCH, iPHONE AND iPAD RUNNING iOS 5.1 OR HIGHER*. Intelligence and Security, Australian Government Department of Defence. March 2012. Available from: http://www.dsd.gov.au/publications/iOS5_Hardening_Guide.pdf, [Accessed online: 30 March 2012].
136. GSM Security. *How Do Authentication and Key generation work in a GSM network?* . Available from: http://www.gsm-security.net/faq/gsm-authentication-key-generation.shtml, [Accessed online: 21 May 2012].
137. Meyer, U. and Wetzel, S., *A Man-in-the-Middle Attack on UMTS*, in *Proceedings of the 3rd ACM workshop on Wireless security (WiSe '04)*. 2004, ACM: Philadelphia, PA, USA. p. 90-97.
138. 3GPP. *Specifications - Confidentiality Algorithms*. Available from: http://www.3gpp.org/Confidentiality-Algorithms, [Accessed online: 1 March 2012].
139. Melia, T., et al. *IEEE 802.21 Mobility Services Framework Design (MSFD)*. Network Working Group. Available from: http://www.rfc-editor.org/rfc/rfc5677.txt, [Accessed online: 7 March 2012].
140. Gruman, G. *Lost in BYOD's uncharted legal waters*. InfoWorld. 6 January 2012. Available from: http://www.infoworld.com/t/byod/lost-in-byods-uncharted-legal-waters-180793, [Accessed online: 14 March 2012].
141. Apple Inc. *iOS Configuration Profile Reference*. 17 October 2011. Available from: https://developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/iPhoneConfigurationProfileRef.pdf, [Accessed online: 5 March 2012].
142. Tindal, S. *IT just like Alice in Wonderland: Defence*. ZDNet. Available from: http://www.zdnet.com.au/it-just-like-alice-in-wonderland-defence-339334268.htm, [Accessed online: 27 March 2012].
143. *How to Clone a SIM Card*. eHow. Available from: http://www.ehow.com/how_4770451_clone-sim-card.html, [Accessed online: 21 May 2012].

144. Sidhardhan, S. *Sim Cloning*. Available from: http://www.5ne.org/sim-cloning/, [Accessed online: 1 March 2012].
145. Scarfone, K. and Padgette, J. *Guide to Bluetooth Security*. National Institute of Standards and Technology. Available from: http://csrc.nist.gov/publications/nistpubs/800-121/SP800-121.pdf, [Accessed online: 2 March 2012].
146. Defense Information Systems Agency. *DoD Bluetooth Peripheral Device Security Requirements*. 16 July 2010. Available from: http://iase.disa.mil/stigs/downloads/pdf/dod_bluetooth_requirements_spec_20100716.pdf, [Accessed online: 30 March 2012].
147. Simonite, T. *Eavesdropping Antennas Can Steal Your Smart Phone's Secrets*. Technology Review, MIT. Available from: http://www.technologyreview.com/communications/39855/page1/, [Accessed online: 16 March 2012].
148. Jun, B. and Kenworthy, G. *Is Your Mobile Device Radiating Keys?* RSA Conference 2012. 2 March 2012. Available from: http://www.cryptography.com/public/pdf/2012-Jun-Kenworthy-MobileDeviceLeakage.pdf, [Accessed online: 21 May 2012].
149. Grubb, B. *'Charlatans and scammers': Googler slams security software firms*. The Age. 23 November 2011. Available from: http://www.theage.com.au/technology/security/charlatans-and-scammers-googler-slams-security-software-firms-20111123-1ntpu.html, [Accessed online: 20 March 2012].
150. Lemos, R. *Apple iOS: Why it's the most secure OS, period*. InfoWorld. Available from: http://www.infoworld.com/print/162792, [Accessed online: 19 March 2012].
151. *Hacker reveals iOS malware vulnerability, gets punished*. GMA News. 10 November 2011. Available from: http://www.gmanetwork.com/news/story/238101/scitech/hacker-reveals-ios-malware-vulnerability-gets-punished, [Accessed online: 20 March 2012].
152. Green Hills Software. *Green Hills Platform for Trusted Mobile Devices*. Available from: http://www.ghs.com/products/mobile_devices.html, [Accessed online: 16 February 2012].
153. *Z800 3DVisor*. eMagin. Available from: http://www.3dvisor.com/, [Accessed online: 5 April 2012].
154. National Security Agency. *TEMPEST FUNDAMENTALS*. published online in redacted form by Cryptome. 6 April 2003. Available from: http://cryptome.org/jya/nacsim-5000/nacsim-5000.htm, [Accessed online: 30 March 2012].
155. Kocher, P.C., Jaffe, J.M., and Jun, B.C., *Differential Power Analysis*. 2009, Cryptography Research Inc.: US Patent No. 7634083 B2.
156. Department of Finance and Deregulation, *A Strategy for the Transition to IPv6 for Australian Government Agencies (Version 2)*. 2009, Australian Government Information Management Office.
157. Catto, B. and Hillier, J., *Australian Government Transition to IPv6*, in *Australian IPv6 Summit Melbourne*. 2011, Australian Government Information Management Office: Melbourne.

158. Davies, E., Krishnan, S., and Savola, P. *IPv6 Transition/Coexistence Security Considerations*. Internet Engineering Task Force. September 2007. Available from: http://tools.ietf.org/html/rfc4942, [Accessed online: 3 April 2009].
159. Hogg, S. *IPv6 Security*. Rocky Mountain IPv6 Summit, Rocky Mountain IPv6 Task Force. 16-17 April 2008. Available from: www.rmcug.org/presentations/RMCUG%20Presentation%20April%202008%20-%20IPv6%20Security.pdf, [Accessed online: 8 April 2009].
160. Abley, J., Savola, P., and Neville-Neil, G. *Deprecation of Type 0 Routing Headers in IPv6*. Internet Engineering Task Force. December 2007. Available from: http://tools.ietf.org/html/rfc5095, [Accessed online: 21 May 2012].
161. National Infrastructure Security Co-ordination Centre, *Security considerations for IPv6*, Technical Note, 26 April 2006.
162. Smith, J. and Morarji, H., *Denial of Service Vulnerabilities in Internet Protocol version 6*, QUT, Draft Report, 2009.
163. Potyraj, C., *Firewall Design Considerations for IPv6*, National Security Agency, I733-041R-2007, 10 March 2007.
164. Chown, T. *IPv6 Implications for Network Scanning*. Internet Engineering Task Force. March 2008. Available from: http://tools.ietf.org/html/rfc5157, [Accessed online: 8 April 2009].
165. Dunmore, M., ed. *An IPv6 Deployment Guide*. 2005, The 6Net Consortium.
166. Convery, S. and Miller, D. *IPv6 and IPv4 Threat Comparison and Best-Practice Evaluation*. 11 March 2004. Available from: http://seanconvery.com/v6-v4-threats.pdf, [Accessed online: 6 May 2009].
167. Conta, A., Deering, D., and Gupta, M. *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*. Internet Engineering Task Force. March 2006. Available from: http://tools.ietf.org/html/rfc4443, [Accessed online: 10 June 2009].
168. Hogg, S. and Vyncke, E., *IPv6 Security*. 1st ed. 2008: Cisco Press.
169. Lucena, N., Lewandowski, G., and Chapin, S., *Covert Channels in IPv6*, in *Privacy Enhancing Techniques*. 2006, Springer Berlin / Heidelberg. p. 147-166.
170. Nordmark, E. *Stateless IP/ICMP Translation Algorithm (SIIT)*. Internet Engineering Task Force. February 2000. Available from: http://tools.ietf.org/html/rfc2765, [Accessed online: 7 May 2009].
171. Tsuchiya, K., Higuchi, H., and Atarashi, Y. *Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS)*. Internet Engineering Task Force. February 2000. Available from: http://tools.ietf.org/html/rfc2767, [Accessed online: 20 May 2009].
172. Lee, S., et al. *Dual Stack Hosts Using Bump-in-the-API (BIA)*. Internet Engineering Task Force. October 2002. Available from: http://tools.ietf.org/html/rfc3338, [Accessed online: 20 May 2009].
173. Hagino, J. and Yamamoto, K. *An IPv6-to-IPv4 Transport Relay Translator*. Internet Engineering Task Force. June 2001. Available from: http://tools.ietf.org/html/rfc3142, [Accessed online: 20 May 2009].
174. Gont, F. *Security Implications of IPv6 on IPv4 Networks*. Internet Engineering Task Force. 27 April 2012. Available from: https://tools.ietf.org/html/draft-gont-opsec-ipv6-implications-on-ipv4-nets-01, [Accessed online: 2 May 2012].

175. Beaumont, M., Hopkins, B., and Newby, T., *Hardware Trojans - Prevention, Detection, Countermeasures (A Literature Review)*, DSTO Technical Note, DSTO-TN-1012, July 2011.
176. Chakraborty, R.S., Narasimhan, S., and Bhunia, S., *Hardware Trojan: Threats and emerging solutions*, in *High Level Design Validation and Test Workshop, 2009. HLDVT 2009. IEEE International*. 2009. p. 166-171.
177. Das, S., Kant, K., and Zhang, N., *Handbook on Securing Cyber-Physical Critical Infrastructure*. 1st ed. Foundations and Challenges. 2012, Waltham, MA: Morgan Kaufmann Publishers.
178. Wei, S. and Potkonjak, M., *Wireless security techniques for coordinated manufacturing and on-line hardware trojan detection*, in *Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks*. 2012, ACM: Tucson, Arizona, USA. p. 161-172.
179. Clark, J., Leblanc, S., and Knight, S., *Risks associated with USB Hardware Trojan devices used by insiders*, in *Systems Conference (SysCon), 2011 IEEE International*. 2011. p. 201-208.
180. Karri, R., et al., *Trustworthy Hardware: Identifying and Classifying Hardware Trojans*. Computer, 2010. **43**(10): p. 39-46.
181. Wang, X., et al., *Hardware Trojan Detection and Isolation Using Current Integration and Localized Current Analysis*, in *Defect and Fault Tolerance of VLSI Systems, 2008. DFTVS '08. IEEE International Symposium on*. 2008. p. 87-95.
182. King, S.T., et al., *Designing and implementing malicious hardware*, in *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*. 2008, USENIX Association: San Francisco, California. p. 1-8.
183. Waksman, A. and Sethumadhavan, S., *Silencing Hardware Backdoors*, in *Proceedings of the 2011 IEEE Symposium on Security and Privacy*. 2011, IEEE Computer Society. p. 49-63
184. Huffmire, T., et al., *Handbook of FPGA Design Security*. 2010: Springer Publishing Company, Incorporated. 177.
185. Skorobogatov, S. and Woods, C., *Breakthrough silicon scanning discovers backdoor in military chip*, in *Proc. of the Workshop on Cryptographic Hardware and Embedded Systems (CHES2012) - to appear*. 2012: Leuven, Belgium.
186. Smart Grid Australia. *Smart Grids in Australia*. Smartgrids Australia. Available from: http://smartgridaustralia.com.au/SGA/Documents/Smart_Grids_In_Australia.pdf, [Accessed online: 24 April 2012].
187. Gibson, D. *The natural gas grid…the really smart grid*. [Blog], American Gas Association. 20 July 2010. Available from: http://www.truebluenaturalgas.org/natural-gas-gridthe-smart-grid/, [Accessed online: 1 May 2012].
188. Top, H. *Smart Grids and Smart Water Metering in The Netherlands*. EC – ICT for Water Management, Acenture. 11 June 2010. Available from: http://ec.europa.eu/information_society/activities/sustainable_growth/docs/water_cons/henk-jan-top_presentation.pdf, [Accessed online: 1 May 2012].
189. Ennesser, F., *The Smart Grid Security Challenges*, in *7th ETSI Security Workshop*. 2012: Sophia Antipolis, France.

190. Office of the National Coordinator for Smart Grid Interoperability Engineering Laboratory, *NIST Framework and Roadmap for Smart Grid Interoperability Standards,*, NIST Special Publication 1108R2, February 2012.
191. Young, S. and Stanic, R. *Study Topic 3: SmartMeter to HAN Communications*. Smart Grid Australia. July 2009. Available from: http://smartgridaustralia.com.au/SGA/Documents/IN_Work_Group_SmartMeter_HAN_Comms.pdf, [Accessed online: 30 April 2012].
192. Maxim Smart Grid Solutions, *Communications*, in *Smart Grid Solutions Guide*. 2011, Maxim Smart Grid Solutions. p. 25-40.
193. 4G-portal.com. *Ausgrid, Australia: Smart grid network to use LTE*. 25 August 2011. Available from: http://4g-portal.com/ausgrid-australia-smart-grid-network-to-use-lte, [Accessed online: 24 April 2012].
194. National Institute of Standards and Technology. *The Role of the Internet Protocol (IP) in AMI Networks for Smart Grid*. Available from: http://www.ietf.org/mail-archive/web/smartpower-interest/current/docFn1Z5XcFuW.doc, [Accessed online: 24 April 2012].
195. Berthier, R., Sanders, W.H., and Khurana, H. *Intrusion Detection for Advanced Metering Infrastructures: Requirements and Architectural Directions*. in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*. 2010.
196. Pollet, J. *Electricity for Free?* The Dirty Underbelly of SCADA and Smart Meters, Red Tiger Security. Available from: https://media.blackhat.com/bh-ad-10/Pollet/BlackHat-AD-2010-Pollet-RTS-Electricity-for-Free-slides.pdf, [Accessed online: 30 April 2012].
197. McLaughlin, S., Podkuiko, D., and McDaniel, P., *Energy Theft in the Advanced Metering Infrastructure Critical Information Infrastructures Security*, E. Rome and R. Bloomfield, Editors. 2010, Springer Berlin / Heidelberg. p. 176-187.
198. Krebs, B. *FBI: Smart Meter Hacks Likely to Spread*. KrebsOnSecurity. 9 April 2012. Available from: http://krebsonsecurity.com/2012/04/fbi-smart-meter-hacks-likely-to-spread/, [Accessed online: 9 April 2012].
199. Anderson, R. and Fuloria, S. *Who Controls the off Switch?* in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*. 2010.

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION
DOCUMENT CONTROL DATA

1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)

2. TITLE

Challenges and Opportunities in Information Security

3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION)

Document	(U)
Title	(U)
Abstract	(U)

4. AUTHOR(S)

Tamas Abraham, David Adie, Angela Billard, Paul Buckland, Samuel Chenoweth, Michael Frangos, Sarath Indrakanti, Martin Lucas, Paul Montague

5. CORPORATE AUTHOR

DSTO Defence Science and Technology Organisation
PO Box 1500
Edinburgh South Australia 5111 Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO-TN-1114	AR-015-382	Technical Note	September 2012

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES	12. NO. OF REFERENCES
2012/1136234/1	INT07/012	DSD, ISG	57	199

DSTO Publications Repository

http://dspace.dsto.defence.gov.au/dspace/

14. RELEASE AUTHORITY

Chief, Command, Control, Communications and Intelligence Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SA 5111

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CITATION IN OTHER DOCUMENTS Yes

18. DSTO RESEARCH LIBRARY THESAURUS

Information Security

19. ABSTRACT

The biennial Infosec Challenges report provides information to the Defence Signals Directorate (DSD) on a range of current and emerging areas in information security. In our 2012 report, areas have been selected to reflect potential information security interests across a broad range of ICT scenarios in the Australian Government. In each of these areas, we consider the current state-of-the-art, in research and/or practice, and identify existing challenges and opportunities.