A Multi-layer Security Model for 5G-enabled Industrial Internet of Things

Hussain Al-Aqrabi[1[0000−0003−1920−7418]], Anju P. Johnson[1[0000−0002−7017−1644]], Richard Hill[1[0000−0003−0105−7730]], Phil Lane[1[0000−0002−2179−0166]], and Lu Liu[2]

[1] School of Computing and Engineering, University of Huddersfield, UK
{h.al-aqrabi,a.johnson,r.hill,p.lane}@hud.ac.uk
[2] Department of Informatics, University of Leicester, UK
{l.liu@leicester.ac.uk}

Abstract. This article considers the need for secure communications between Industrial Internet of Things (IIoT) devices that operate in 5G-enabled environments. 5G networks enable greater data throughput and lower latency, which presents new opportunities for the secure authentication of business transactions between IoT hardware. We propose an approach to developing a flexible and secure model for IIoT components in 5G environments. Using the NIST seven layer model of cloud computing, in conjunction with Physically Unclonable Function (PUF) hardware provided via FPGAs, we demonstrate algorithms that replicate common authorisation challenges in 5G enabled IoT scenarios.

Keywords: Internet of Things · 5G · security · Physical Unclonable Functions · analytics

1 Introduction

Adopting emerging business models that can exploit the potential of Internet of Things (and Industrial Internet of Things) devices, is tempered by valid security and privacy concerns [1–3]. Industrial users in particular recognise that a significant proportion of the value that they generate is directly linked to the ownership and continued development of Intellectual Property (IP). Any breach of security that might threaten the exclusivity of IP presents a considerable risk upon the underlying business model of an enterprise [4].

Whilst cloud computing is an example of how technologies and operating models can come together to offer new capabilities for businesses [7, 2], there are still cases of organisations being threatened by new risks which are attributed explicitly to the adoption of cloud services [8, 9, 14].

5G infrastructure promises new opportunities for closer integration of physical devices [17] which is particularly suited to IIoT as follows:

– *Greater data throughput* facilitates the exchange of data between devices, which supports the introduction of meta data to support robust transactions to establish trust between devices;

- *Lower power requirement* permits the extended use of sensing and computational devices in remote settings;
- *Reduced network latency* improves response times and has the potential to improve the rigour of security protocols, without harming the user experience of the system.

The use of millimetre band for 5G is a key enabler of the improved network performance, albeit at a loss of transmission distance. While the higher throughput frequency band provides some physical security [24, 25], this cannot be relied upon. A malicious insider stood next to machinery could relay data outside [18, 20, 19, 16]. Therefore, we need to be able to develop a scalable security architecture that can robustly marshall the appropriate authentications for various parties, whilst being tolerant of the dynamic nature of how IIoT devices can interact.

This article describes how a cloud approach has been used to inform the design of a security model for the following reasons. First, cloud architectures directly support dynamic demands through elasticity [13, 12, 15], and promote the harmonisation of disparate technologies through abstraction. Second, there is a mature architectural reference model provided by NIST [11], and this is universally adopted. Third, cloud systems exhibit shared characteristics with IIoT systems, in that there are multiple parties that need to collaborate [7] and cooperate through the secure sharing of data and resources, generally using a service-based approach.

Prior work [4, 5, 7, 6] has explored the particular case of multi-party authentication in cloud based architectures, for the provision of enterprise Business Intelligence systems [8, 6, 9]. We have developed and extended this work to support the development of specific use cases where the availability of 5G network infrastructure can enable new business possibilities through enhanced performance. To enable these possibilities, we have augmented a cloud-based architecture to include hardware with Physically-Unclonable Functions (PUF).

PUFs are resistant to spoofing attacks, which is important for situations where it is necessary to rapidly authenticate a party, or multiple parties, to establish trusted communications. The provision of analytics services from manufacturing plant is a use case that we have considered, where there is a need for the secure exchange of raw data, as well as processed results, between IIoT components and the organisation's enterprise system.

This work has considered probable attack vectors upon such a system, to assist in the design of an effective multi-layer security model. In doing so, we have developed algorithms to allow authentication using PUFs, to enable secure access to cloud-based applications. As such, we consider the emerging scenario whereby IIoT devices are being attached and embedded within manufacturing plants to facilitate new levels of secure coordination, control, knowledge exchange, and ultimately, the protection of Intellectual Property provided by fundamental business operations.

The remainder of this article is organised as follows. In Section 2, we describe multi-layer security models. Section 3 describes a multi-layer approach

to connectivity using Physical Unclonable Functions (PUF). In Section 4, we provide experimental results to establish the effectiveness of the design. Finally, we conclude in Section 5.

2 Multi-layer Security Model

A key challenge of managing network connectivity in a densely-equipped environment such as a smart factory, is the identification and marshalling of different attack vectors. Using a fundamental principle of cloud computing - resource abstraction - we have elected to construct a security model that separates a number of checkpoints across multiple layers. Again, the convenience of the cloud computing model has inspired us to consider each of the layers as a discrete cloud which will contain myriad hardware resources such as servers, storage arrays, FPGA devices and suchlike. As such, system designers are free to focus optimisation efforts on each cloud as a separate entity to ensure that operating overheads are kept within acceptable timeframes [23].

Figure 1 illustrates the overall security model architecture. We have continued with the use case of a typical enterprise system that contains analytics functionality for the purposes of supporting strategic and operational business decision-making.

As such, our model has been tested in a cloud environment where individual users are tenants within a multi-tenant scenario. Since the model allows for resource abstraction, users making requests for access to the enterprise system can do so via Virtual Machines, remote systems, and directly via hardware devices such as IIoT components. For the purposes of modelling within Figure 1, each user or IIoT device is represented as an eventual tenant of the multi-cloud enterprise system.

All external requests are initially marshalled by firewalls that hold authentication data for each of the prospective tenants. This metadata contains access controls for the functionality that has already been authorised for each individual tenant. As the first line of defence, a lack of requisite authentication data will prohibit a user from successfully connecting to the system.

After basic authentication has been verified, there is a Tenant Metadata layer, which holds the rule-based controls to indicate which aspect of the enterprise system can be accessed by each authorised tenant of the system. This might refer to specific repositories, or reporting for example. In the case of IIoT devices providing data for analytics processing, this is likely to include not only the appending of data to a repository, but also the controlled access to other data sources that may be fused and aggregated to with the process data, in order to provide enhanced analytics services.

It is necessary to establish a trustworthy connection and this is achieved through the use of PKI. Public Key certificates are retained within a Digital Vault, within its own layer of the model, and this provides a further level upon which a tenant (user) session can be authorised or eliminated.

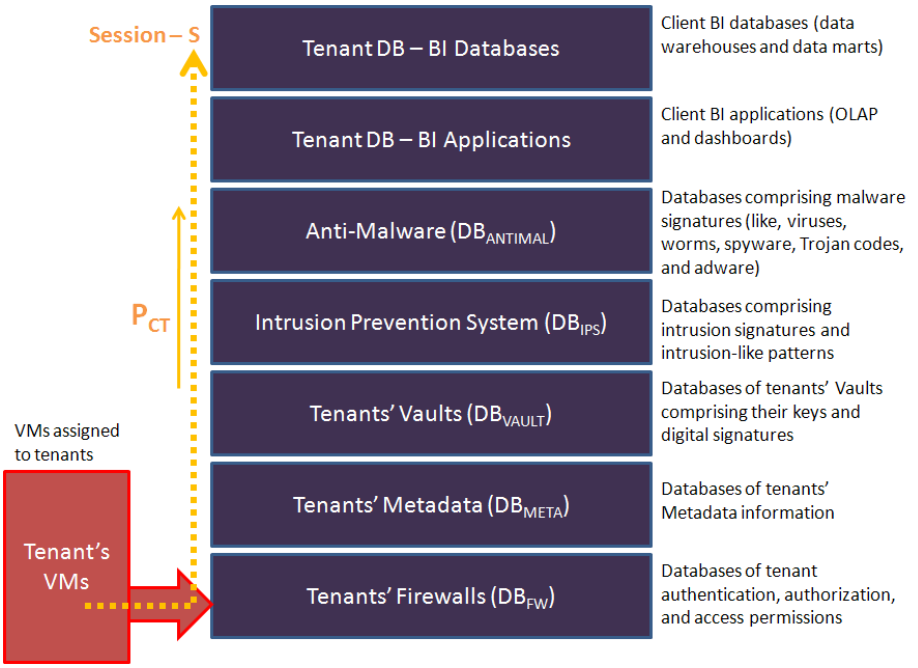

Fig. 1. Layered security model [5] for an organisational enterprise system inspired by NIST[10].

Layer four provides a deeper level of protection for instances where malicious attackers have successfully penetrated the first three layers. While the previous layers' controls are sufficient to protect against many threats, they do not protect against a malicious insider who will already have suitable credentials to gain access to the system so far. The Intrusion Prevention System contains logic to identify patterns and strange behaviour, in order to conclude the session of a user who is engaging in threatening behaviour.

Layer four is bolstered by an anti-malware protection layer. More surreptitious activity, such as concealed executable code, can create havoc when deployed further into the enterprise system. Layer five maintains a log of such activity, as well as a repository of threats that have been identified.

The applications layer cloud contains the enterprise functionality that is of interest to the business users. By the time a user has arrived at this layer, basic authentication, session authentication via PKI, intrusion detection and anti-malware checks have already been made, with opportunities at each layer to terminate a session. Within the application layer, there is still the need for role-based permission control, so that data of differing sensitivity can be accessed by the appropriate authority.

Aside from enterprise applications, there is a need for a particular type of user to have access to enterprise repositories, either directly via APIs, or through

querying and reporting interfaces, typically delivered through a web portal. Layer seven provides for such access, and similar to the preceding layers, abstracts the functionality into a cloud for additional protection.

2.1 NIST seven layer model

The security model can be mapped to the NIST seven layer model of cloud computing [10] as follows. Tenant users, which can be either VMs or hardware devices, are resident in NIST layers $1-3$.

Each session is usually marshalled to layer 6 (if the destination is a Software-as-a-Service, SaaS, application) through a series of authentication and verification steps contained within the fourth and fifth layers. Access to layer 7 is reserved for API interfaces to applications that are likely to be hosted on premises, and are usually unique to the organisation.

Using cloud nomenclature, the firewall is an example of Infrastructure-as-a-Service (IaaS) and subsequent controls are Platform-as-a-Service (PaaS) with the exception perhaps of level 6 which may be SaaS.

2.2 Session workflow

As described above, the security model maps to the IaaS and PaaS layers of the NIST model for cloud architecture [10].

The instantiation of a new session by a prospective IIoT tenant user, results in session IDs being assigned in layers two and three. This is followed by access identification provided from within layer four. It is after this stage that packet inspection is the fundamental activity for each of the sessions that have progressed thus far.

The initial $\mathrm{DB}_{\mathrm{META}}$ and $\mathrm{DB}_{\mathrm{VAULT}}$ layers enable IIoT requests to be verified, prior to packet inspection for each session to be performed using $\mathrm{DB}_{\mathrm{IPS}}$ and $\mathrm{DB}_{\mathrm{ANTIMAL}}$. In the context of the NIST model, $\mathrm{DB}_{\mathrm{IPS}}$ and $\mathrm{DB}_{\mathrm{META}}$ map directly to PaaS functions. In contrast $\mathrm{DB}_{\mathrm{FW}}$ is regarded as IaaS in the model.

If there is a SaaS instance at layer six, there will be additional authentication applied for each user, though at this stage, a considerable amount of verification has taken place already. However, this authentication check is to enforce enterprise system role-based permissions, for organisational data protection, such as the subset of staff who require access to confidential payroll information.

3 Multi-layer connectivity model using PUF

So far, our security model accomodates requests from users for access to services (typically analytics) in an industrial setting. We are primarily concerned with user requests eminating from hardware IIoT devices which are either embedded or retro-fitted to manufacturing plant.

Such devices may be represented by FPGA units, that include storage and computational resource that can be used to process data at the edge of enterprise

networks. This functionality has a great deal of potential for industrial use cases if the data collected and processed locally, can be combined with additional enterprise data in real-time. The low-latency characteristics of 5G networks are thus attractive solutions to such challenges.

The multi layer model allows for elasticity in the system, which can tolerate not only an increase in processing volume, but also an increase in the total volume of transactions in response to an increase in the number of IIoT devices that are communicating with the system.

We now consider the approach taken to introduce an IIoT component that itself has design features that increase the level of trust between devices, through the use of Physically-Unclonable Functions (PUF).

The first protocol to consider is the process by which a new user is enabled to access an IIoT node, after being introduced to the system by an existing user.

There exist K verification layers in the model. For every genuine client IIoT device (FPGA in this example), there exists a PUF in each layer of our security model (we consider a $K = 7$ layer cloud model). The PUF is used for authentication and is unique to each genuine client. The cloud management unit generates a combined PUF model (M_A) (mathematical) which is transferred through a secure communication channel.

The model is represented as an obfuscated bitstream which is downloadable and implementable on the client FPGA. An authorisation request of U_A is handled by the management plane, which sends q challenge bits CH_p, each of length n and a random number $rand$ to U_A.

U_A applies the challenge bits on the mathematical model and generates the responses for each layer. There exist K responses for a single challenge string. Hence a total of $K.q$ responses are generated, which are jumbled using a pre-agreed shuffling scheme with a random number $rand$.

The shuffled string of responses, encoded using a pre-agreed encoding scheme $E(.)$ is sent to the security model for verification. The encoded responses are decoded using the decoding scheme $D(.)$ and are reordered by the management layer and are provided to the K layers for verification.

At each security layer, the original challenge bits are applied to the respective PUFs in the cloud and the received model responses are compared with the physical PUFs. Upon successful authentication at all K layers, U_A is declared to be trustworthy. We consider a strong arbiter PUF with > 10 constituent XOR-PUFs to resist any machine learning attacks [24, 30, 31].

3.1 Multi-layer hierarchical packet inspection using PUF

Central to the overall secure connectivity model is the use of multi-layer hierarchical packet inspection. Each IIoT client FPGA must satisfy some standards to qualify as a valid recipient of the cloud service.

Each FPGA requires a Dynamic Partial Reconfiguration (DPR) capability and dynamic partitions in the FPGA fabric to promote analysis by the cloud service. We use (PUF) in addition to the existing verification to ensure security. Each of the cloud layer FPGAs contains PUFs representing each client. We

Algorithm 1 Multi-layered connectivity model using PUF: Client is an existing User

Objective:

(a) The seven layer cloud model consisting of FPGA clouds verifies the identity of a client FPGA (U_A) who is requesting access.
(b) The cloud model provides application access for the genuine client (U_i).

Prerequisites:

(a) An n-bit input, 1-bit output XOR PUF P_1 is reconfigured in all layers of the $Cloud-FPGA$. There exists a PUF for every authenticated user. PUF P_{ij} represents the identity of the user i in the cloud layer j.
(b) A combined mathematical model M_i representing all the K PUFs in the cloud layers, resides with each user U_i.
(c) $Cloud-FPGA$ and user U_i have agreed on a fixed encoding scheme $E()$ and a decoding scheme $D(.)$, such that for any binary string x, $E(.)$ and $D(.)$ are injective, $X = E(x)$ and $D(X) = x$.
(d) $Cloud-FPGA$ and user U_i have agreed on a shuffling scheme $Y = S(X, rand)$, and $S'(Y, rand) = X$ where $rand$ is a random number.

Input:
S, P_{CT}, DB_{FW}, DB_{META}, DB_{VAULT}, DB_{IPS}, $DB_{ANTIMAL}$

(a) Tenant session: S
(b) Contents of session packets:P_{CT}
(c) Contents of FW: DB_{FW}
(d) Contents of $TENANT_{META} : DB_{META}$
(e) Contents of $TENANT_{VAULT} : DB_{VAULT}$
(f) Contents of $IPS : DB_{IPS}$
(g) Contents of $ANTIMALWARE : DB_{ANTIMAL}$
Note: DB_j represents content DB of layer j

assume that amn IIoT client contains a mathematical model of the PUF which is downloaded as an obfuscated bitstream to the client FPGA using DPR. A strong PUF which cannot be cloned using machine learning strategies is of paramount importance to ensure the overall security of this model.

The client FPGA sends PUF responses of the mathematical model to the multi-layer cloud model for verification. Side channel parameters are collected by the malware detection layer by downloading an obfuscated bitstream on the client FPGA. This is to ensure that the IIoT client FPGAs do not produce any parameter variations following infected hardware trojans [25].

This fingerprint measured in terms of side-channel parameters is a means of identifying and preventing attack by hardware trojan horses in the device. Multiple side channel parameters are collected using dynamic partial reconfiguration.

Algorithm 1 continued. Output:
A value in variable S to show that the application access is granted ($S = 1$) or denied ($S = 0$).

Steps:

1. Initialize $S = 1$, $E = 1$
2. U_i to management plane MP: request access to application A
3. MP to U_i: MP sends a random number $rand$ and a set of challenges CH_p consisting of q challenge bits each of length n.
4. U_i calculates the following:
   - $Rim_{p,j} = Mi(CH_{p,j})$, $p = 1 \dots q$, $j = 1 \dots K$
   - $Rim = \{Rim_{p,j},\ 1 \leq p \leq q,\ 1 \leq j \leq K\ \}$
   - $CA_i = S\Big(E(Rim), rand\Big)$
5. U_i to MP: certificate CA_i
6. **foreach** layer j **do**
   (a) Initialize $Mem = 0$, $Match = 0$
   (b) **If** $(E = 1)$
      (a) $MP : Rim_{p,j} = S'\Big(D(CA_i), rand\Big)$
      (b) MP to $Cloud - C_i$: Set of challenges CH_p and $Rim_{p,j}$
      (c) $Cloud - C_j$ calculates the following
         - $Rif_{p,j} = Pi(CH_{p,j}), p = 1 \dots q, j = 1 \dots K$
         - $N_{ij} = (1 - \frac{\sum_{(p=1)}^{q} (R_{imp} \oplus R_{ifp}}{q})$
         - **if** $Nij \geq 0.99$ $Mem = 1$
      (d) **if** $(P_{CT} \in DB_j, \mid DB_j \in \{DB_{FW}, DB_{META}, DB_{VAULT}\}$ AND $P_{CT} \notin DB_j, \mid DB_j \in \{DB_{IPS}, DB_{ANTIMAL}\}\ )$; $Match = 1$
      (e) **if** $(Mem\ \&\&\ Match)$, $E = 1$; proceed to next higher layer
      (f) **else** Exit; set $E = 0$, $S = 0$; DenyTenantAccess()
7. **if** $S = 1$; AuthoriseTenantAccess()

The bitstreams are erased using another DPR following successful collection of samples. The cloud directly collects samples to avoid the client from manipulating the parameters.

3.2 Algorithm design

Any user requesting access to an application is described in Algorithm 1. Each item in the sequence represents the various checks that are enacted through the multi-layer security model. We have provided for extensibility through this design, in that should an enterprise require additional security layers to be augmented, such controls can be implemented using the basic premise of abstraction to marshall and control the findings of any stage of packet inspection.

One example that is particularly pertinent to manufacturing enterprises arises through developments in thinking around Industry 4.0 or *smart manufacturing*. These movements advocate the sharing of resources and services in a horizontal

fashion, cutting across industrial sectors to facilitate new levels of collaboration. This thinking is driven by new capabilities in analytics, and the control and feedback of Cyber Physical Systems (CPS). In such cases, orgainsations that choose to embrace the sharing of services, and indeed exploit the possibilities of packaging existing functionality into services to share, will require the ability to be able to extend the security model so that satisfactory security inspections are in place.

Algorithm 1 thus describes an arrangement where a set of challenge bits are sent to the client and the cloud layers. The management unit responds with a set of q challenges, each of size r.

For every user, the management unit maintains a database of challenges that have been previously used for PUF-based authentication, and disregards every challenge that has expired.

However, for challenge set size of q in each authentication attempt, with each challenge being r-bit, the probability of repeated challenges occurring (represented as the "Failure Probability") is almost zero with a typical value of $q = 1000$ and $r = 64$ [23]. Each IIoT client generates model responses and sends them to the cloud security layers for verification.

Cloud layers then generate responses using the physical PUF. Both responses are checked for a match of at least 99%. Following a successful match, the content of the packet is analysed by the cloud before proceeding to the next cloud layer in the security model.

Algorithm 2 represents the steps for introducing a new IIoT client to the application. In this case, an existing IIoT client introduces a new IIoT client by sharing its model responses.

On successful authentication of the existing client, the model generates a new PUF in the dynamic part of the cloud FPGA fabric. A combined mathematical model of the PUFs is downloaded to the client FPGA (as before) using an obfuscated bitstream following a DPR process. Again, the security model assumes that a secure DPR process is followed to retain system integrity. The new cloud client then utilises Algorithm 1 to provide access to the application layer.

4 Hardware Implementation

The basic XOR PUF architecture consisting of 64 challenge bits and 10 parallel arbiters is implemented on an FPGA to compute the design expense in a cloud architecture. Nexys 4 DDR board with Artix-7 FPGA [27] (device xc7a100t, package csg324, speed -1) from Xilinx is used for implementation.

The architecture is designed in Verilog Hardware Description Language (HDL) following the flow in Xilinx ISE 14.7 design suite [26]. Xilinx power analysis tool [28] and Chipscop-Pro [29] are used for testing and analysis.

Table 1 reports the implementation cost in terms of resources consumed, total on-chip power (using vector-less activity propagation) and the configuration bitstream size.

Algorithm 2 Multi-layered connectivity model using PUF: New client

Objective:

(a) The seven layer cloud model consisting of FPGA clouds verifies the identity of a new client FPGA (U_B) who is requesting access.
(b) The cloud model provides application access for the genuine client (U_B).

Prerequisites:

(a) New client $ClientU_B$, requesting application access is known to an existing client U_A as a genuine applicant
(b) $Cloud - FPGAs$ have built-in controllers to facilitate secure dynamic partial reconfiguration.
(c) $Client - FPGA$ has built-in controllers to facilitate secure dynamic partial reconfiguration initiated by the cloud.
(d) The $Cloud - FPGA$ fabric is divided into two parts, a) static fabric and b) dynamic fabric. Static fabric consists of hardware configurations which existed before deployment. The dynamic fabric of the $Cloud - FPGA$ is dedicated to configure additional security primitives (mostly PUFs) for any genuine clients using secure dynamic partial reconfiguration.
(e) The $client - FPGA$ fabric is divided into two parts, a) static fabric and b) dynamic fabric. Static fabric consists of hardware configurations which existed before deployment. The $Client - FPGA$ has secure remote DPR controllers in the static partition facilitating configuration of PUF mathematical model in the dynamic fabric, via an obfuscated bitstream.

Input:
P_{CT}, DB_{FW}, DB_{META}, DB_{VAULT}, DB_{IPS}, $DB_{ANTIMAL}$ of $UserU_A$

(a) Tenant session: S
(b) Contents of session packets:P_{CT}
(c) Contents of FW: DB_{FW}
(d) Contents of $TENANT_{META} : DB_{META}$
(e) Contents of $TENANT_{VAULT} : DB_{VAULT}$
(f) Contents of $IPS : DB_{IPS}$
(g) Contents of $ANTIMALWARE : DB_{ANTIMAL}$
Note: DB_j represents content DB of layer j

Output:
A value in Flag to show a successful dynamic partial reconfiguration ($Flag = 1$) or denied ($Flag = 0$).

Steps:

1. Initialize $V = 1$, $E = 1$, $Flag = 0$
2. U_B requests U_A, for an introduction to access application A
3. U_A to MP: request introduction of U_B to cloud layers C_j
4. MP to U_A: MP sends a random number $rand$ and a set of challenges CH_p consisting of q challenge bits each of length n.
5. U_A calculates the following:
   - $RAm_{p,j} = MA(CH_{p,j})$, $p = 1 \dots q$, $j = 1 \dots K$
   - $RAm = \{RAm_{p,j},\ 1 \leq p \leq q,\ 1 \leq j \leq K\ \}$
   - $CA_A = S\Big(E(RAm), rand\Big)$
6. U_A to MP: certificate CA_A
7. **foreach** layer j **do**
   (a) Initialize $Mem = 0$, $Match = 0$
   (b) **If** $(E = 1)$
      (a) $MP : RAm_{p,j} = S'\Big(D(CA_A), rand\Big)$
      (b) MP to $Cloud - C_j$: Set of challenges CH_p and $RAm_{p,j}$
      (c) $Cloud - C_i$ calculates the following
         - $RAf_{p,j} = PA(CH_{p,j}), p = 1 \dots q, j = 1 \dots K$
         - $N_{Aj} = (1 - \frac{\sum_{(p=1)}^{q}(R_{Amp} \oplus R_{Afp}}{q})$
         - **if** $NAj \geq 0.99$ $Mem = 1$
      (d) **if** $(P_{CT} \in DB_j, \mid DB_j \in \{DB_{FW}, DB_{META}, DB_{VAULT}\}$ AND $P_{CT} \notin DB_j, \mid DB_j \in \{DB_{IPS}, DB_{ANTIMAL}\}\ )$; $Match = 1$
      (e) **if** $(Mem\ \&\&\ Match)$, $E = 1$; proceed to next higher layer
      (f) **else** Exit; set $E = 0$, $Flag = 0$
8. **if** $V = 1$; Verified introducing client
   (a) **foreach** layer j **do**
      (a) $Cloud - FPGA$, C_j initiates DPR and configures a new PUF $P_{B,j}$, PUF $P_{B,j}$ represents the identity of the U_B in the cloud layer j
      (b) C_j to MP PUF modeling parameters $param_j$
   (b) MP generates a combined Mathematical model M_B of all PUFs $P_{B,j}$ in the cloud layers
   (c) MP generates obfuscated bitstreams of PUF mathematical model M_B
   (d) MP initiates remote dynamic partial reconfiguration of PUF M_B in the dynamic partition of the $client - FPGA$ U_B
   (e) $Flag = 1$ and exit; follow protocol-1. U_B is same as any other existing client.

Results show that the XOR PUF is realissable with reduced resources and occupies only 8% of the device slices. This figure is negligible for large FPGAs deployed in clouds and data centers.

From Table 1, it is evident that the required bitstream size for adding a reconfigurable PUF is relatively small. The Internal Configuration Assess Port (ICAP) of the targeted FPGA board is designed to run in 8 bit configuration at a clock frequency of $100MHz$. Hence DPR for the required PUF file is performed in the order of micro-seconds.

Table 1. Implementation Overhead

Hardware Consumption*	Slice	Slice Reg	LUTs
	1291	10	1282
Power Consumption	$0.082W$		
Bitstream Size	$3737KB$		

*Note: The design does not contain any LUTRAMs, BRAMs/FIFOs, DSPs or buffers

5 Conclusions

We have described the extension of a prior work on multi-layer cloud security models [5] to include hardware devices with PUF capability. Using experience of cloud architectures and myriad attack vectors upon Virtual Machines in multi-tenant cloud systems, we describe a PUF-based system that demonstrates an ability to marshall not only external adversarial attacks, but also internal attacks, which are particularly of relevance to industrial organisations.

The use of cloud-inspired abstraction layers leads to an architecture where a wide range of surreptitious activity can be quarantined to resist a multitude of attack vectors. Multiple layers of packet inspection enusures that determined adversaries, who may employ a number of methods simultaneously, can be halted.

Security is provided beyond firewalls by compromising further attacks and therefore establishing a trail of evidence of the potential multitude of exploit attempts. An IPS layer prevents one level of exploit, and is shored-up by an anti-malware layer that detects other forms of compromise such as trojans.

A key advantage of this approach is the inherent flexibility and scalability that is brought about by the cloud-inspired layering and containerisation of security functions. The continued expansion and accessibility of IIoT hardware, together with ever-stronger business cases for embedding analytics functions into industrial organisations, means that extensible, flexible architectures are required to take full advantage of high speed networks and increased numbers of IIoT devices.

We have proposed the use of hardware security primitives such as PUFs to ensure security within the model. This is strengthened by monitoring of the side channel parameters of client IIoT devices to provide additional security from hardware trojans. The robustness of the implementation relies upon this integration of both software and hardware technologies to ensure security in cloud computing in the IIoT revolution.

References

1. S. Kalra and S. K. Sood. Secure authentication scheme for IoT and cloud servers. Pervasive and Mobile Computing, Elsevier, 24, 2015, pp210223.
2. A. Kadan. Security Management of Intelligent Technologies in Business Intelligence Systems, Proceedings of International conference on Modeling and Simulation, Minsk, Belarus, May 2-4, 2012, pp70-72.
3. A. Alrawais, A. Alhothaily, C. Hu, X. Cheng. Fog computing for the Internet of Things: Security and privacy issues, IEEE Internet Comput., 21(2), pp34-42, Mar 2017.
4. H. Al-Aqrabi, L. Liu, J. Xu, R. Hill, N. Antonopoulos, Y. Zhan. Investigation of IT Security and Compliance Challenges in Security-as-a-Service for Cloud Computing, Proceedings of 15th IEEE International Symposium on Object/component/service-oriented Real-Time distributed Computing (ISORC2012), Shenzhen, China, April 11-13, IEEE, pp. 124-129.
5. H. Al-Aqrabi, L. Liu, R. Hill, N. Antonopoulos, A Multi-layer Hierarchical Inter-Cloud Connectivity Model for Sequential Packet Inspection of Tenant Sessions Accessing BI as a Service. Proceedings of 6th International Symposium on Cyberspace Safety and Security and IEEE 11th International Conference on Embedded Software and Systems. France, Paris, March 20-22, IEEE, 2014 pp. 137-144.
6. H. Al-Aqrabi, L. Liu, R.Hill, N. Antonopoulos. Cloud BI: Future of business intelligence in the Cloud, Journal of Computer System Science, Elsevier, 2015.
7. H. Al-Aqrabi, R. Hill. Dynamic Multiparty Authentication of Data Analytics Services within Cloud Environments. In 20th IEEE International Conference on High Performance Computing and Communications (HPCC-2018), Exeter 28-30, IEEE.
8. S. Ouf, et al. Business Intelligence Software as a Service (SaaS), Proceedings of 3th International Symposium on Communication Software and Networks. Xian, 27-29 May 2011, IEEE, pp641-649.
9. Z. Panian. How to Make Business Intelligence Actionable through Service-oriented Architectures. WSEAS Transactions on Business and Economics, 5(5), pp210-221, 2008.
10. NIST. US Government Cloud Computing Technology Roadmap. Special Publication 500-293. Cloud Computing Program, National Institute of Standards and Technology (NIST), US Department of Commerce, 2011, pp13-78.
11. The NIST definition of cloud computing, Peter Mell, Tim Grance, Computer Security Division, Information Technology Laboratory, NIST, 2011
12. Y. Demchenko, C. D. Laat. Defining Generic Architecture for Cloud Infrastructure as a Service Model, Proceedings of 4th International Symposium on Cloud Computing Technology and Science (CloudCom), Taipei, Taiwan, December 3-6, 2012, pp2-10.
13. Y. Demchenko, C. Ngo, C. Laat, D.R. Lopez, A. Morales, J.A. Garca-Espn. Security Infrastructure for Dynamically Provisioned Cloud Infrastructure Services, in book Privacy and Security for Cloud Computing, Springer London, pp167-210.

14. R. Hill, P. Lake, L. Hirsch, S. Moshiri. Guide to Cloud Computing, Springer Verlag, 2013, ISBN 978-1-4471-4602-5.
15. M. Carvalho. SECaaSSecurity as a Service, ISSA Journal, October 2011, pp20-24.
16. S. Luo, et al. Virtualization security for Cloud computing service. Proceedings of 4th International Symposium on Cloud and Service Computing, Hong Kong, December 12-14 2011, IEEE, 174-179.
17. D. Shadija, M. Rezai, R. Hill. Towards an Understanding of Microservices. Proceedings of the 23rd International Conference on Automation & Computing, University of Huddersfield, 7-8 September, 2017, IEEE.
18. N. Kumar, L. DuPree. Protection and Privacy of Information Assets in the Cloud, Book Chapter: Auditing Cloud Computing: A Security and Privacy Guide, Halpert, B. (Ed.), NY: Wiley, 2011.
19. S. Pearson, Taking Account of Privacy when Designing Cloud Computing Services. Proceedings of International Symposium on Software Engineering Challenges of Cloud Computing, CLOUD '09, Vancouver, Canada, May 23- 23, 2009. IEEE, pp1-9.
20. D. Diaz-Sanchez, et al. Media Gateway: bringing privacy to private multimedia cloud connections, Telecommunication Systems, 2013, Springer, pp1-16.
21. Z. Pervez, A.M, Khattak, S. Lee, Y. Lee. SAPDS: self-healing attribute-based privacy aware data sharing in Cloud, Journal of Supercomputing, 62, 2012, Springer, pp431-460.
22. D.W. Chadwick and K. A, Fatema. A privacy preserving authorisation system for the Cloud, Journal of Computer and System Sciences, 78, pp13591373, 2012.
23. Johnson, Anju P., Rajat Subhra Chakraborty, and Debdeep Mukhopadhyay. "A PUF-enabled secure architecture for FPGA-based IoT applications." IEEE Transactions on Multi-Scale Computing Systems 1.2 (2015): 110-122.
24. Tobisch, Johannes, and Georg T. Becker. "On the scaling of machine learning attacks on PUFs with application to noise bifurcation." International Workshop on Radio Frequency Identification: Security and Privacy Issues. Springer, Cham, 2015.
25. Narasimhan, Seetharam, Dongdong Du, Rajat Subhra Chakraborty, Somnath Paul, Francis G. Wolff, Christos A. Papachristou, Kaushik Roy, and Swarup Bhunia. "Hardware Trojan detection by multiple-parameter side-channel analysis." IEEE Transactions on computers 62, no. 11 (2012): 2183-2195.
26. ISE Design Suite Overview, Xilinx Inc,[Online]. Available: www.xilinx.com/support/documentation/sw_manuals/xilinx11/ise_c_overview.htm, Accessed: 2019-05-21.
27. Digilent Nexys 4 Artix-7 FPGA Trainer Board, Xilinx Inc, [Online]. Available: www.xilinx.com/products/boards-and-kits/1-3yznp5.html, Accessed: 2019-05-21
28. Xilinx Power Estimator User Guide, Xilinx Inc, 2018, [Online]. Available: www.xilinx.com/support/documentation/sw_manuals/xilinx2018_3/ug440 -xilinx-power-estimator.pdf, Accessed: 2019-05-21
29. ChipScope Pro Software and Cores,Xilinx Inc, 2012,[Online]. Available: www.xilinx.com/support/documentation/sw_manuals/xilinx14_7/chipscope_pro_sw_cores_ug029.pdf, Accessed: 2019-05-21
30. R. Pappu, B. Recht, J. Taylor, N. Gershenfeld, 2002. Physical one-way functions. Science. 297 (5589): 20262030. Bibcode:2002Sci...297.2026P. doi:10.1126/science.1074376. hdl:1721.1/45499. PMID 12242435.
31. B. Gassend, D. Clarke, M. van Dijk, S. Devadas. Silicon Physical Random Functions. Proceedings of the Computer and Communications Security Conference, November 2002.