A HIGH ASSURANCE FIREWALL IN A CLOUD ENVIRONMENT USING HARDWARE AND SOFTWARE

Dr. Arya Golriz,
Nur Jaber

Faculty Advisors:
Dr. Richard Dean, Dr. Yacob Astatke and Dr. Farzad Moazzami

Department of Electrical and Computer Engineering
Morgan State University

ABSTRACT

This paper will focus on analyzing the characteristics of firewalls and implementing them in a virtual environment as both software- and hardware-based solutions that retain the security features of a traditional firewall.

INTRODUCTION

With the evolution of telemetry into network environments, telemetry experts must consider the impact of cloud computing solutions in future telemetry systems. Cloud computing is becoming increasingly popular and offers a wide variety of advantages over conventional networking, including the ability to centralize resources both physically and financially. While implementing a cloud infrastructure does raise security concerns, a secure cloud infrastructure similar to that of a conventional network can be achieved using tools and tactics deployed to protect the network from adversaries and various malicious attacks.

One primary component in any secure network, cloud or otherwise, is a firewall that examines inbound and outbound traffic on the network to ensure that it is authentic and based on a set of rules, as well as enables the network administrator to permit safe content. A cloud infrastructure differs from a conventional network mainly in its logical implementation, so building a secure cloud network will differ logically. Finding the best combination of a virtual firewall and its implementation is instrumental to building a fast, efficient cloud network that also has all the properties of a secure conventional network. The virtual firewall's effectiveness will be measured according to existing federal standards and definitions regarding network security.

Telemetry systems are becoming more virtualized, hence creating security vulnerabilities. This paper focuses on addressing a lack of boundaries in these virtualized environments. In a traditional network, boundary defense is trivial by comparison since physical boundaries are implemented between users and network components, such as firewalls and guards. But the transition to a virtual network eliminates these physical boundaries because, with a cloud, most network components essentially sit on a single server.

While a firewall between users and the Internet can be implemented similarly to that of a conventional network, the defense needed between users poses a problem. Traditionally, a firewall can protect users from one another because it is able to monitor traffic traversing the internal network, but a physical firewall in the cloud is essentially blind to traffic between virtual machines because the separation between users is logical.

This paper will demonstrate how to reinstate the boundary between users in the cloud by dedicating physical resources, processors in the host, which will establish a physical separation between each virtual machine.

Dedicating processing power to the firewall ensures that overloading it with innumerable tasks will not diminish the firewall's ability to address potential threats because, typically, processing power is idle until summoned by various tasks. Cloud providers generally are large corporations that employ massive data centers and command servers with plenty of processing power; however each user is given an allotment of CPU usage per virtual machine. Amazon Web Services, for example, charge extra for CPU On-Demand instances per hour when users want to exceed their plan.

Dedicated processors to the firewall will provide the needed separation between users and also provide a cost-effective measure to keep the user from exceeding allotted CPU usage, since the firewalls should always have a high priority and be slated to run in the event of an attack.

This measure provides security assurance, as highlighted in 20 Critical Security Controls, where the NIST references the Boundary Defense objective to detect, prevent, and correct the flow of information transferring networks of different trust levels and focusing on security-damaging data. In this case, since users do not know or trust one another, each one will be treated as networks of different trust levels.

BACKGROUND

A suitable design criteria is needed for this approach. Several levels of security are defined in the NIST FIPS 140-2document [1] of computer security standards to accredit cryptographic modules. While this paper does not address the use of cryptography as a solution, the standards for implementing a secure infrastructure are relatively similar. They consist mainly of physical security features that must be implemented to account for potential insider/outsider threats trying to gain access to either other users' machines or servers that rely on access to the cloud:

- Security Level 1: Basic, low level security requirements that allow software and firmware components to be executed on a general purpose computing system using an unevaluated OS.
- Security Level 2: Requires authentication of a user.
- Security Level 3: Prevents unauthorized users from gaining access to critical security parameters by creating physical or logical separation between users.
- Security Level 4: Physical protection from environmental factors.

While FIPS 140-2 is a standard for security of a cryptographic module, it can be used as a guideline to establish many different security standards, including establishing a secure virtual network infrastructure. The four different levels of security specified by FIPS 140-2 are a clear indication of this. The various levels of security indicate that there must be separation of components both logically and physically. This paper mainly focuses on providing Levels 2 and 3 protection. While Levels 1 and 4 are certainly present, they do not constitute the primary focus of concern because no cryptography is required.

All these requirements are typical of a secure network infrastructure and can be applied to establishing a secure standard for a firewall. The different requirements that FIPS 140-2 specifies are module specification; ports and interfaces; roles/services/authentication; finite state model; physical security; operational environment; cryptographic key management; self-tests; design

assurance; and mitigation of other attacks. All are applicable to the firewall except cryptographic key management.

If all these requirements can be met for any security module, then the module has a level of high assurance. The security module being proposed in this paper is able to satisfy all the requirements set forth by FIPS 140-2 and, therefore, can be classified as a secure, high-assurance security module.

APPROACH

Current virtual firewalls primarily operate on an application level using available OS processing power, which pose concerns about limited resources and scalability. This study will demonstrate the advantages of using a software- and hardware-based solution to meet all the needs of a user in a cloud environment. While the firewall will be deployed on each individual virtual machine, dedicated processing cores will alleviate concerns about the potential lack of processing power regarding firewalls.

This paper addresses the physical issue by proposing a dedicated security processing core for each VM that is online. An example would be a VM with an octo-core processor on the cloud, with one core dedicated to the firewall and the remaining seven cores dedicated to all remaining processes. Where there is high CPU usage by a particular user, the firewall would be unaffected by any and all of these processes. This solution would prohibit the user from overloading and causing crashes that could shut down processes like the firewall and expose the network to threats.

In order to build an effective test, it is important to understand not only network security, but also operating system security. The cloud is hosted by a server that runs an OS of its own. With multiple users accessing shared resources, an OS is considered to be a multiprogrammed OS, also known as a monitor. [2] Protecting each user on the OS is accomplished by implementing one of four types of separation:

- Physical Separation: Different processes use different physical objects for output implementing different levels of security.
- Temporal Separation: Processes employ different security requirements and are executed at different times.
- Logical Separation: Users operate with the impression that no other processes exist. For example, an OS limits a program's access so that it cannot operate outside its permitted domain.
- Cryptographic Separation: Processes hide their data and computations so they are meaningless to outside processes.

Firewall solutions for a virtual network to the Internet are not a primary concern in this study because the current infrastructure is more reliable than the internal network. This is especially important in a commercial cloud where users want to depend on it as though it is their own private network. There are two Ethernet adapters for each VM; one for the internal network, and one for Internet access.

The design intention is to dedicate to the firewall to a CPU for security purposes to avoid a potential bottlenecking issues. A verification check, using the FIPS 140-2 standard that outlines security of cryptographic modules, can be performed to validate that the firewall is functioning properly.

In addition to partitioning CPU on single chips to dedicate security, another method of achieving the security goal is to dedicate an entire processor to the firewall. The Application-Specific Integrated Circuit (ASIC) chip is a processor, by design, that is dedicated to a single process.

Traditional firewalls suffer from low performance because all security functions are performed through a single CPU. They also only serve a single function, which means additional network security devices must be purchased and implemented to achieve network security.

The application-specific integrated service (ASIC)-based firewall – a dedicated custom-made chip developed for custom-made security functions that also solves the bottleneck issue. The ASIC-based firewall takes care of data transmission and execution of security functions while the CPU takes care of various configuration tasks, excluding handling, collection of statistical data, and user interface. Dedicated security chips are easily stacked, which increases performance. In addition, custom-made chips are very difficult to alter, eliminating the distrust users may have in the OS.

The firewall structure being proposed combines these functions using a software- and hardware-based solution. Data channels demand for processing power is fulfilled with dedicated hardware, while control protocol's flexibility is accomplished with the general-purposed CPU. The firewall will be divided into an ASIC and a general-purpose high-performance CPU. The ASIC will accomplish main security functions, especially data applications at the TCP layer and above, and the general purpose CPU will run system software. This implementation in the cloud server will help establish a physical boundary, allow up-to-date firewall definitions, provide efficient performance, and negate the need to assume a trust level of the OS. [3]

In order to maintain the idea of assigning CPU to the firewall, the ASIC can be used to provide both physical and logical separation and solve the security bottleneck issue. As a dedicated custom-made chip developed for custom-made security functions, it takes care of data transmission and executing security functions, while the CPU takes care of various configuration tasks, exception handling, collecting statistical data, and user interfaces.

The ASIC optimizes system performance, and makes efficient use of board space because it does not incur the overhead of fetching and interpreting stored instructions. The ASIC enhances the performance of traditional processors, taking care of data transmission and executing security functions. It's enhanced design security makes it virtually impossible to reverse engineer.

To provide assurance that the firewall is functional, techniques will be used that involve trusted third party markings of network packets to facilitate automated forensics such as flagging packets. The trusted third party system will serve as a guard to allow network entities to communicate. A security module in the guard will ensure that each incoming and outgoing packet has first gone through the firewall. Network processors will insert trustworthy security marks in the IP header of incoming packets, which should be applied to all the packets forwarded since the third party guard is a priori unaware of potentially malicious intent of any packets it forwards.

The IP header currently has 33 available bits that can be used to append security marks. As it stands, these are the available header bits: Identification – 16 bits; Fragment Offset – 13 bits, Type of Service – 2 bits; More Fragment – 1 bit; Fragment Flag – 1 bit. Because of complications in the various header fields, it has been determined that the ID field or Type of Service fields are the best candidates for a security flag. When a packet passes through VM 1, it must first go through ASIC Firewall 1, which sets the security flag bit to "1" once it determines

that the packet is safe and contains no malicious content. If the packet is deemed unsafe, it is dropped. If the packet is harmless, it receives the security flag.

After passing through the firewall, it must go to Guard 1 where the security module sits. If it is determined that the security flag has been not been set, VM 1 will shut down with a notice sent to the security officer. If it determines that the flag has been set, then the packet is permitted passage to Firewall 2. Firewall 2 determines if the packet is being received from a reliable source and whether it contains malicious content. If the packet is trusted, a second security flag is set. If not, it is dropped.

The packet will then goes through Guard 2, which checks for the two security flags. If both flags are not set to "1", VM2 will shut down and a notice will be sent to the security officer. If both flags have been set, then the packet will be permitted passage to VM2. Figure 1 shows the firewall security module.

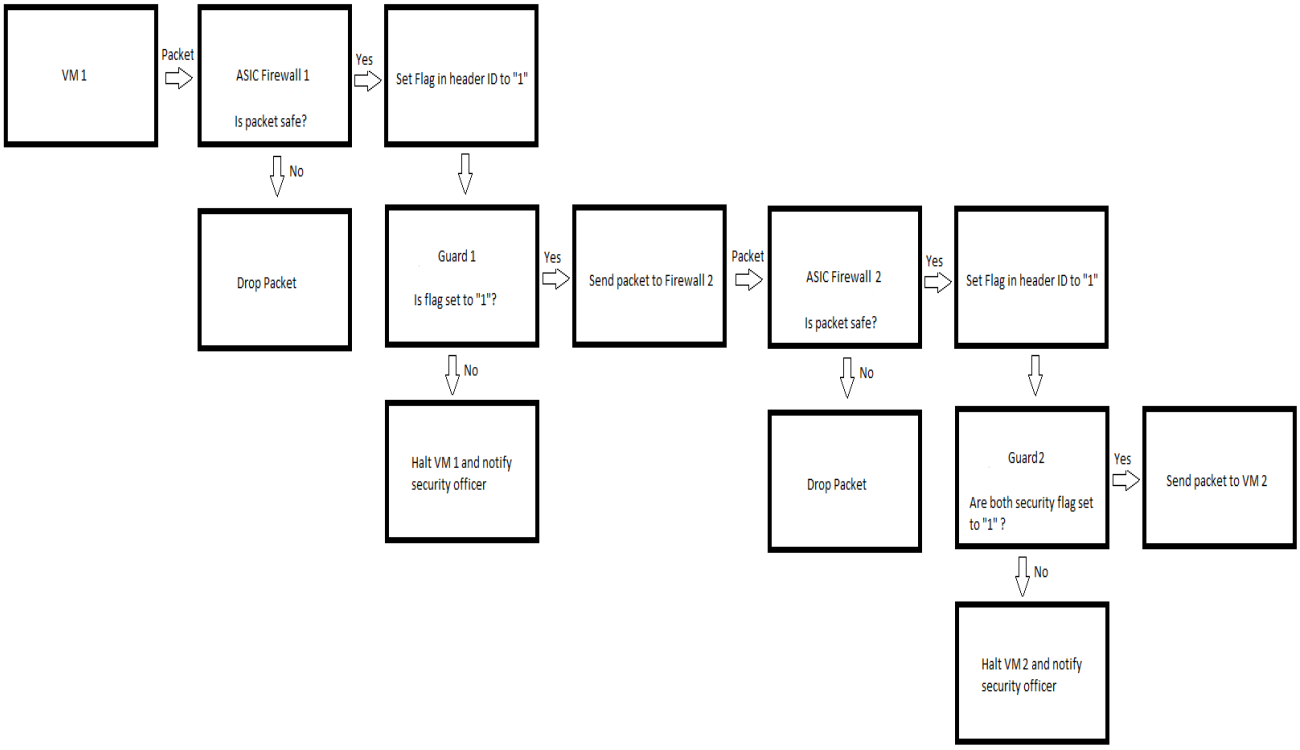

Figure 1: Firewall Operational Assurance Mechanism

The third party guard provides assurance that the firewall is actually functional, adding a second measure of security. This security assurance cannot be spoofed since any packet coming out of the firewall must physically pass through the guard. In providing a solution that protects the boundaries in the cloud, it is important to distinguish where those boundaries are physically. Figure 2 portrays the virtual network in which the security module can be seen.

The virtual local area network (VLAN) on the host machine contains the cloud's virtual machines. The security modules, which contain the ASIC firewall and third party guard, also sit

on a tray in the host where they are assigned to each VM joining a new session by the dynamic firewall module switch. The firewall then belongs to that user session until the user terminates the VM connection.

The only physical connections are between the firewall, guard, and dynamic switch. All remaining connections are virtual, other than those that are required by the hardware resources within the host machine. All the virtual machines are logically in the same network, with the security switch allocating module resources to each new user that joins the network. Depending upon need, the security module bus can be connected internally or externally. An Ethernet connection has been chosen for this design based on its ability to transmit data rapidly. A copper or fiber Ethernet connection can be used, but fiber, with its higher data rate, is used in this design.

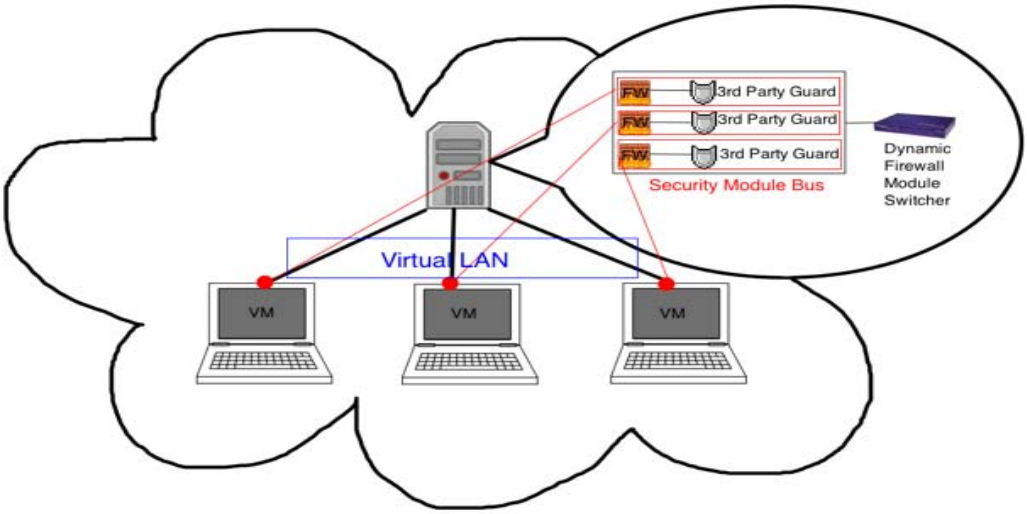

Figure 2: Firewall security module in the Cloud

The security module contains the bus, which consists of the ASIC firewall and third party guard, as well as the dynamic firewall module switch. Also included is a mechanism that allows for firewall updates. Internally, the ASIC firewall board is connected to the third party guard with an Ethernet cable. The host is then connected to the switch using an Ethernet cable.

Since the ASIC firewall has a completely separate, simple OS that is designed specifically for its use, updates to firewall policies and definitions cannot be altered or pushed through the network by outsiders. Updates or alterations must be done internally and manually because the secure framework cannot be altered. The design includes an update module on the firewall board that is connected to Core 2 of the CPU, the task core, when updates need to be performed. At all other times, it is left disconnected. Updates can be performed by a USB connection or other type of removal media. This ensures that the firewall definitions cannot be modified unless an attacker has physical access to the host.

Figure 3 shows the connectivity between the internal components of the security module, as well as the connectivity between the module and the host. The security module contains the

bus, which consists of the ASIC firewall and third party guard, as well as the dynamic firewall module switch. Also included is a mechanism that allows for firewall updates. Internally, the ASIC firewall board is connected to the third party guard with an Ethernet cable. The host is then connected to the switch using an Ethernet cable.

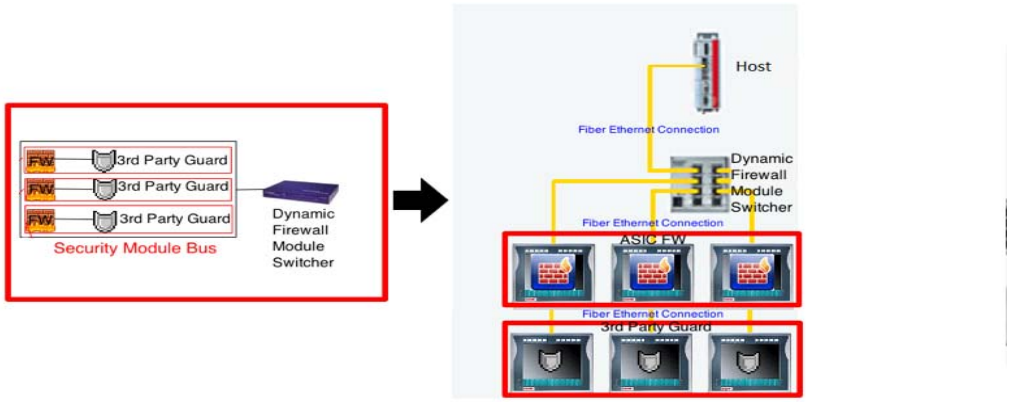

Figure 3: Bus structure of ASIC firewall security module

Since the ASIC firewall has a completely separate, simple OS that is designed specifically for its use, updates to firewall policies and definitions cannot be altered or pushed through the network by outsiders. Updates or alterations must be done internally and manually because the secure framework cannot be altered. Figure 4 conceptually shows how the security module is designed to allow for this type of update.

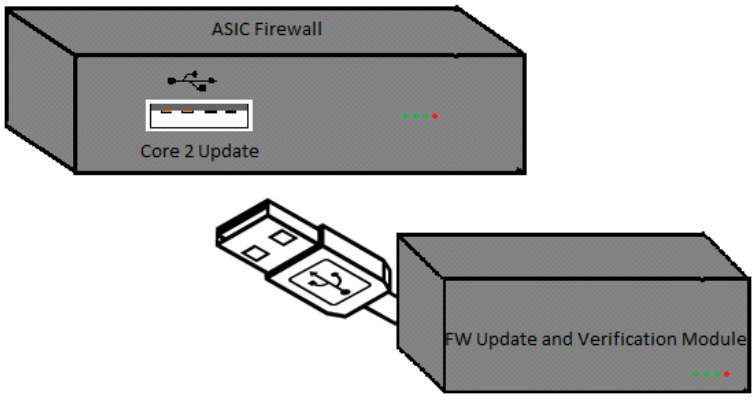

Figure 4: Security Module Design for Updates

Self-tests, conducted at the beginning of a session and periodically throughout it, provide assurance that the firewall is intact and operational. Performed by the security module, they

ensure that the firewall can send and receive packets, append a security bit, send them to the guard for verification, and send them back to the firewall for transmission. Figure 5 shows a flow chart of how this is achieved in this design.

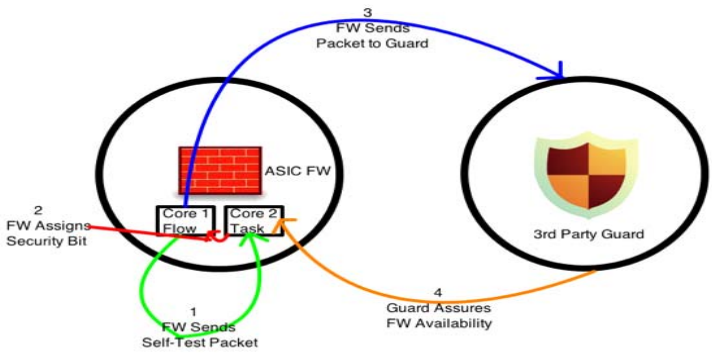

Figure 5: Security module self-test

In the first step, the firewall flow core transmits a small self-test packet to the firewall task core. Then, Core 2 verifies the packet and appends a security bit. The packet is then sent to the third party guard for verification. Upon verification of the security bit, the third party guard ensures that the firewall is intact and available, and then transmits the packet back to the task core. Once the packet is received by the task core, the self-test is complete.

Should the self-test fail, the VM is halted and the security officer is notified, which is similar to the firewall failing during a normal packet transmission.

When there is no user activity, it is recommended that the firewall perform periodic self-tests approximately every 30 seconds. Should any step fail, the entire system reports a failure and shuts down the VM. If Core 2 is waiting to receive the packet to complete step 4 and has not received a packet within 30 seconds, the system concludes that the security module has failed and notifies the security officer. If a test packet is expected and fails to arrive, the system has most likely been compromised. Any failure in one of the steps triggers the alarm. The security officer receives these reports periodically, as well. If nothing is reported, then the firewall has failed and the VM must be manually shut down.

Checking each of the specifications with the design of the firewall assurance mechanism in place verifies that it closely follows the FIPS 140-2 standard. Levels 2 and 3 of the standard are invoked to provide role-based authentication, i.e., the security officer; and production-grade components, such as the ASIC-based firewall, third party guard, and security module switch.

Level 3 security is achieved by the safety stickers on the devices, although this is not shown in this paper. Level 4 security is also present, but it must be achieved by the network administrators who assemble the network and provide environmental safety features.

CONCLUSIONS AND FUTURE WORK

Since cloud computing is the logical implementation of a traditional network, it may seem that it holds the same security standards. However, this paper demonstrates that, regarding security, this is not always the case. Perhaps the greatest vulnerabilities in the cloud are the most fundamental ones – separation and boundaries. While firewalls between cloud users and the Internet are relatively simple to implement, the boundaries between users pose a problem. Users in a cloud share resources, so there is no physical separation between them.

When using FIPS 140-2 and implementing a firewall, it was also necessary to provide assurance that the firewall was, in fact, operational in case of possible attack. This was achieved by flagging incoming and outgoing packets in the firewalls and checking them with third party guards. Devising an ASIC-based virtual firewall provides administrators with the ability to make changes to policy without making hard changes to the actual firewall, since ASIC chips are nearly impossible to alter. The third party guard is able to provide needed assurance by performing incoming and outgoing packet checks, restrict traffic, support user and officer roles, demonstrate the ability to transition the VM to on/off states and notify officers, physical security, depending on a simple and specifically designed OS, and can perform self-tests.

After implementing these design features and performing a series of logical tests, it was determined that this design provides both the functionality and assurance needed to secure a cloud by providing protection between users. Installing boundaries between users assures a first line of defense and makes it difficult for a network to be compromised from within.

Future work may include the physical design of this ASIC-based firewall with an assurance mechanism. Building the ASIC firewall would include the processors assigned to flow and task as well as a chip to play the role of the third party guard to provide assurance. Future work should also include the implementation of this security mechanism on small and large scales to ensure reliability at all levels. The cost and difficulty of doing so, however, is beyond the scope of this paper. However, it would certainly be the next step of building a secure cloud infrastructure.

REFERENCES

[1] National Institute of Science and Technology. (2001, May) Federal Information Processing Standards 140-2.

[2] Charles P. Pfleeger, *Security in Computing, Fourth Edition*. USA: Prentice Hall , 2006.

[3] SifoWorks. Firewall For the Next Generation.

[4] National Security Agency, Information Assurance Technical Framework Release 3.0, September 2000.

[5] Pawan Kumar, "Analysis of Different Security Attacks in MANETs on Protocol Stack A-Review ," *International Journal of Engineering and Advanced Technology* , vol. 1, no. 5, pp. 270-275, June 2012.

[6] Cloud Security Alliance. (2010, March) Top Threats to Cloud Computing V1.0. [Online]. https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf