https://doi.org/10.48009/2_iis_2014_242-251

USING LAYERING TO ENHANCE THE SECURITY OF AN E-COMMERCE SYSTEM DESIGNED TO SUPPORT GLOBAL BUSINESS

Dennis Guster, Saint Cloud State University dcguster@stcloudstate.edu
Paul Safonov, Saint Cloud State University, safonov@stcloudstate.edu
Andrea Hathaway, Saint Cloud State University, haan0808@stcloudstate.edu

ABSTRACT

The literature indicates that cloud computing is an exciting and advantageous architecture, well suited to support global business due to its scalability needs, but it also involves new and substantial security risks. This paper provides an overview of a configuration of a cloud that was implemented in the authors' laboratory and was broken into three layers: the cloud, zone, and virtual machine. Security mechanisms that were used in the study, including multi-level firewalling, are described. A series of actual port security definitions within each firewall are included so that the interaction among the three firewalls can be illustrated and analyzed. Finally, we conclude by discussing the importance of starting with a sound security policy before configuring the firewalls.

Keywords: Information Technology (IT), Cloud Computing, Cloud Security, Layering

INTRODUCTION

Cloud computing provides an excellent architecture to support large-scale applications and ensure both high reliability and excellent performance. However, because it is a relatively new architecture there are significant risks from a security perspective [1]. While data encryption is an integral part of any security strategy because of the segmentation, isolation and inheritance strategies used in a cloud, other tools such as layering are critical to ensure data is not compromised [13]. A cloud is a multi-dimensional computing environment that is broken into zones and virtual machines (VM), and a potential user would likely have different rights in different parts of that cloud, but this complexity can make it difficult to track security issues [19]. This situation leads to being able to filter bad data on multiple levels. For example, it would be easy to implement a firewall on the cloud, the zone and the VM level, so therefore a potentially dangerous data packet could be examined three times and hopefully would get filtered out on at least one level. The same is true with services, for instance web sites in different zones on different VM may have different security profiles. Because of this complexity many IT professionals feel that the full ramifications of cloud computing and its security challenges are still unknown [12]. It is important to remember that a cloud is primarily a virtual world. In fact, the hardware for a small cloud may only involve 5 physical computing units, but with multiple cores per box. In the authors' case those five units each contain 24 cores, so the cloud contains 120 computing cores. However, virtualization often increases the number of logical computers (virtual machines) to a value in the hundreds. This scenario promotes green computing by reducing hundreds of hosts to a platform of 5 physical hosts, which results in major savings in power and cooling. However, for the security officer the logical model is still very complex and can involve numerous levels of abstraction through the use of virtual zones, virtual hosts, replication and virtual networks. Because of this small footprint, company managers often do not recognize the "logical" complexity and embrace a security strategy that is "disruptively simplified" [3].

Some security issues are unique to a cloud while others are present potentially on any host supporting services. Often a service that is compromised on a cloud magnifies the potential damage that could occur. With the concept of global services contained within the cloud, damage that before would only effect the target host might be extended to the entire cloud. This is especially true if single sign-on is supported via the cloud's active directory authentication structure (LDAP) [11]. Therefore, there are shared security issues between cloud and traditional computing and it is important to understand that a dangerous security issue in traditional computing could become worse in a cloud environment [17]. However, there are potential security advantages in cloud computing as well. In other words, it is easier to isolate and add multiple protection levels to vulnerable services. However, for the "layering approach" to be effective it is important that security policy and documentation be updated to reflect the characteristics of this new architecture. In fact, many IT professional have stated that a whole new paradigm is needed to deal with the added complexities of cloud computing [22].

It is clear that there are numerous security issues in a cloud that merit further investigation. While traditional computing security solutions rely on a layered approach, cloud computing can vastly improve that strategy by adding additional layers. So therefore the primary objective of this work is to illustrate how additional layers of security can be implemented within cloud architecture. Specifically, this paper hopes to zero in on such topics by focusing on the layering of firewall logic within the entire cloud. Because a firewall is such a basic security tool one would expect that there is policy in place to guide the filtering of data for each service and its corresponding port. Too often this policy is geared to protect resources primarily from incoming attacks. However, outgoing traffic should also be a concern, particularly that which might get out of a VM that is compromised. Conventional logic might dictate that a VM within the cloud should be viewed as a trusted entity and rights on that VM be inheritable by other VMs with in that cloud. In practice it is safer to treat each VM as potentially corrupted and not allow specifically defined traffic between or among the internal VMs to take place. To accomplish this, the traffic entering and exiting the cloud would be protected by a firewall. However, a second layer or zone would be created within the cloud. For example, a class C IPV4 license could be sub-netted, creating two network zones. Perhaps one zone would be for production activities and the other for research and development activities (R&D). Once again, traffic within the cloud could be filtered so that R&D traffic could not penetrate the production zone. This filtering could also be extended to the VM level. For example, a VM supporting accounting functions within the production zone could become inaccessible from a VM supporting inventory related services. This paper will perform an audit of a cloud containing multiple zones and multiple virtual machines within the authors' cloud, delineate the design strategy and evaluate the effectiveness of a layered firewall strategy. Because the experiment is being undertaken in a test bed the results are subject to the following limitations:

1. The processors used are PC style X86-64.
2. The cloud architecture was built using VMware.
3. The virtual machines were built with the Ubuntu Linux operating system.
4. The security tools (such as UFW) used are part of the release of the Ubuntu operating system.
5. The total number of processor available was limited to 120.
6. The number of zones configured in the cloud was limited to two.
7. The multiprocessor algorithm used was SMP (symmetric multi-processing).
8. The global file system used is NFS (network file system).
9. The active directory used is LDAP (light-weight directory access protocol).

REVIEW OF LITERATURE

Advantages of Clouds

As one might expect, the literature explains that there are many advantages to cloud computing, but many risks as well [1]. Because of the newness and complexity of this architecture many feel that unknown security vulnerabilities may be the greatest obstacle to the adoption of cloud services [4]. For many it is difficult to conceptualize the degree that data can be isolated due to a cloud's logical nature and less reliance on physical separation [21]. On the contrary, if properly configured a cloud can offer improved multi-level isolation when compared to traditional physical hosting solutions [13].

Multi-layer Security Approach

Because of a multi-layer security approach coupled with the use of global services, the full ramifications of the unique security concerns of this architecture need further evaluation [12]. Global services are useful in illustrating security concerns that are typical in cloud architecture [16]. Specifically, in a cloud, users do not login to a host. Instead, users login from a centralized user database, such as LDAP (lightweight directory access protocol). This scenario allows single sign-on capability to any host in the cloud. However, if the LDAP is compromised, instead of gaining access to a single host, the hacker may have gained access to the whole cloud. Therefore, it is important to assign specific security levels to users and limit their access to a certain zone or VM within the cloud [17]. This problem could also be addressed by employing a trusted third party, much like certificates. The trusted third party would use public key encryption in concert with SSO (single sign on) and LDAP (lightweight directory access protocol) to ensure the authentication, integrity and confidentiality of the system was assured [23]. Of course, this authentication methodology could be supplemented on the data level by using one of the latest public key encryption algorithms [18]. However, the unique aspects of cloud computing, particularly with layering and inherited trust issues, complicate the delivery and deployment models utilized and therefore a sound authentication policy alone will not completely solve this problem [20]. Isolation is critical and a good example for the need of logical isolation is side channel attacks [14]. Specifically, in this methodology a hacker obtains a virtual machine (VM) on the same physical host as the VM of the target within a commercial cloud. The hacker then mines the common physical memory to obtain authentication information that in turn he or she will use to log into the intended target. The key concept here is that attacks such as these show that traditional perimeter security approaches will not be adequate [14]. Therefore, it is critical to have a layered approach and put up logical boundaries throughout the cloud to limit not only what gets in and out of the cloud, but within the zones and VMs within the cloud as well.

Service Security in Clouds

Devising a security strategy for cloud computing can be directly related to defining security metrics, and ensuring the evaluation of those metrics is possible by means of sound logging across the entire cloud. For example, cloud computing can support a wide variety of applications. Specifically, an IaaS (Infrastructure as a Service) client needs to be assigned security based on its application level and the log files for those applications need to be used to evaluate security effectiveness [5]. Once again, because of layering the log files may be located in several areas of the cloud and represent several VMs. It has been suggested due to the complexity of a cloud that one of the services running in that cloud needs to directly deal with security issues. This idea has been termed “policing as a service” [22]. In addition to providing data safe guards, this service attempts to empower the user by providing interactive monitoring capabilities [22]. This concept matches current cloud design fundamentals that emphasize centralization of vital services [15]. Prior to the widespread use of cloud computing, the hacker’s primary attack strategy shifted from attacks on the network and the transport layer to the application layer, which makes it even more critical that security within the cloud address all layers. In summary, the user will be running his or her application on somebody else’s hardware using somebody else’s software, which makes deploying a sound cloud computing security strategy paramount [9].

Clouds in Global Business

As one might expect, adapting the cloud to support global business results in magnifying security issues within the cloud. Specifically, the literature indicates that there are security**,** privacy, and related regulatory issues which will need to be prioritized, managed, and mitigated before full implementation can comfortably occur [6]. Furthermore, the literature states that the basic advantages of a cloud transfer well in supporting a complex application such as global business. For example, the use of cloud computing technology might result in a reduction in major software investments and removal of major infrastructure-related tasks such as system backup, disaster recovery and system management**.** In terms of specific applications, global transactions and the use of a supply chain are the business areas that may benefit the most from cloud computing because the implementation may result in efficiency and cost savings [7]. The literature also recognizes that global business supported by cloud computing will require a robust security strategy. Security particularly becomes problematic when the applications are housed on a public cloud. In any case, whether a public or private cloud is involved, a balanced approach is suggested that implements sound security across multiple levels [8]. The literature also delineates the importance of cloud computing to support the future growth of global business. It is important that security experts devise and enable an effective multi-layer security approach within cloud computing, since it will play a vital role in allowing the global scalability that is required for the successful implantation of cloud technologies [2].

DESIGN OF THE CLOUD AND ZONES

One of the primary tenants of a security strategy is the use of layering [4]. In other words, the logic can be explained by using the analogy of a medieval castle. There was a wall around the entire castle if that was breached then there were inner walls and each zone had multiple level of protection such as draw bridges, archers and boiling oil. Therefore, a successful attacker had to breach many layers. The same basic concept can be applied to a cloud design. The first layer of defense typically is the cloud level firewall. This firewall is designed to filter out bad traffic and can do it based on a network.node address, a specific port or based on the characteristic of a packet. As we will see it is possible to place multiple firewalls within the cloud. They can be placed to protect a specific subpart of the cloud (zone) or a virtual machine in the cloud. So with this layering approach a hacker might have to circumvent three firewalls to successfully attack his/her target. Once on the target host a hacker may still have to compromise other layers of security. For example, the service itself may have security mechanisms in place. In the case of an e-commerce system running on http, the apache webserver supporting the http traffic can be configured to be more secure via its configuration file. This file can typically be viewed by: cat /etc/apache2/apache2.conf. In this file specific directives such as timeout intervals, server threads and lock files can be set which provide extra protection from denial of service attacks. The concept of layering can even be extended to where the executables and data are stored to support the e-commerce system. In this case, three layers are common: Protection on the file system, directory and file levels. For example, the settings that ultimately protect the apache version 2 configuration file are displayed below.

```
/dev/sda1 on / type ext4 (rw,nosuid,errors=remount-ro)
drwxr-xr-x 7 root root    4096 Mar 25 06:28 apache2
rw-r--r-- 1 root root 8346 Feb  6  2012 /etc/apache2/apache2.conf
```

The first line describes what is permitted on the file system level, while read and write access are permitted by default if the file system is compromised then it is remounted in read-only mode. Further, a parameter is set to prevent inheritance of super-user rights from other file systems. The directory level as defined in the second line above prevents write access by anyone except the “root” user, even the group “root” has no write access. The last line provides protection on the file level and once again limits write access to the user root. From this analysis it is clear that there are many layers a hacker must defeat to take advantage of data/services within a cloud.

To further illustrate the security issues associated with using a cloud to support global business activities, the authors configured a test-bed within their instructional laboratory. From a hardware perspective, this cloud resided on five computing units each containing 24 computing cores resulting in a cloud with 120 cores. Network connectivity was provided with an enterprise level switch supporting up to 10 Gbs line speed. Network layer access to the cloud was provided via a Class C IPV4 license. To illustrate the concept of zones this license was sub-netted so that the lower half of the node range was assigned to cloud management functions such as DNS (domain name service), global authentication via LDAP and replication of global data via NFS. The upper half of the address space was in turn allocated to support global business activities. Of course, the primary services would be http/https which are needed to support e-commerce. The kernel routing table from a VM within the upper half zone illustrates the mask required to implement the sub-netting. This strategy could be easily expanded if it was desired to break the cloud into more zones. The current mask is 25 1's (the 1's indicate the network portion). If the mask were changed to 255.255.255.192, it would become 26 1's and allow the cloud to be broken into 4 zones. Note that throughout the paper the true IP addresses are not used for security purposes. Rather, the public class C address is used as a place holder.

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window irtt Iface
default         192.19.59.254   0.0.0.0         UG        0 0          0 eth0
192.19.59.128   *               255.255.255.128 U         0 0          0 eth0
```

Although both zones are contained within the same hardware, they are logically isolated from one another. The host within the upper half has the following IP address.

```
fred@globus:~$ hostname -i
192.19.59.190
```

While the host shows up in a DNS search, it is isolated and cannot be pinged from the other zone.

```
fred@globus:~$ dig lowerhalfaddress

;; ANSWER SECTION:
lowerhalfaddress. 3600   IN      A       192.19.59.14

fred@globus:~$ ping -c5 192.19.59.14
PING 192.19.59.14 (192.19.59.14) 56(84) bytes of data.

--- 192.19.59.14 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4011ms
```

An isolation strategy can also be applied on the service level. In the example below http/https are available from anywhere (* = wildcard) while DNS records must be pulled from within the cloud using the VM itself (either the internal domain name inl.local or localhost). Further, to remotely administer this host an administrator would have to have access to the appropriate zone in the cloud and the VM named globus before access to the administration service webmin would be allowed.

```
tcp        0      0 *:http                  *:*                     LISTEN
tcp        0      0 *:https                 *:*                     LISTEN
tcp        0      0 globus.inl.local:domain *:*                     LISTEN
tcp        0      0 localhost:domain        *:*                     LISTEN
tcp        0      0 *:ssh                   *:*                     LISTEN
tcp        0      0 globus.inl.local:webmin *:*                     LISTEN
```

It is also possible to provide an additional layer of security on the data itself. The example below shows how the main static portion of the website (http) above can be better protected by placing it at a remote site and setting it to read only, not allowing files to be executed and stopping inheritance of super-user IDs.

```
fred@globus:/var/www$ ls
index.html
remotedatastore.inl.local on /var/www type nfs4 (ro,noexec,nosuid)
```

Using Log Files to Access the Isolation Strategy

By using the UFW (uncomplicated firewall) software (as opposed to iptables) it is easy to setup and evaluate firewalls. This is especially important in a cloud because there will be many firewalls implemented. In the example below, the basic premise is illustrated on a VM level firewall. The host globus is running telnet (just to illustrate how isolation can save your ASCII). Tom is on another host within the cloud and the same zone attempts to connect to no avail. The firewall log shows that the connection request was indeed blocked at the firewall level. Note the destination port is in fact port 23 which is assigned to telnet.

```
fred@globus:/var/log$
tcp        0      0 *:telnet                *:*                     LISTEN

tom@host2.inl.local:~$ telnet 192.19.59.190
Trying 192.19.59.190...
telnet: Unable to connect to remote host: Connection timed out
May  5 11:19:20 globus kernel: [8529104.969716] [UFW BLOCK] IN=eth0 OUT=
MAC=00:50:56:8c:52:33:00:50:56:8c:45:a8:08:00 SRC=192.19.59.234 DST=192.19.59.190
LEN=60 TOS=0x10 PREC=0x00 TTL=64 ID=38637 DF PROTO=TCP SPT=48098 DPT=23 WINDOW=14600
RES=0x00 SYN URGP=0
```

To truly illustrate the isolation capabilities of cloud computing a simple test-bed was configured within the authors' laboratory. This configuration featured three layers of firewalls: the cloud, zone and VM layers.

Firewall Layering

For the sake of simplicity let's assume the following: Firewall on the edge of cloud: 192.19.59.1.
Firewall for the upper half of the address range: 192.19.59.234.   Firewall on the VM: 192.19.59.190

192.19.59.1

```
To                           Action      From___________
22/tcp                      ALLOW IN    Anywhere
22/tcp                      DENY  IN    172.19.0.0/16
40811/tcp                   DENY  IN    Anywhere
44444/ucp                   DENY  IN    192.19.25.1/24 (v6)
80/tcp                      ALLOW IN    Anywhere
443/tcp                     ALLOW IN    Anywhere
514/udp                      DROP        Anywhere
514/udp                      ALLOW IN    192.19.59.0
```

192.19.59.234

```
To                           Action      From___________
22/tcp                       ALLOW IN    Anywhere
40811/tcp                   ALLOW IN    192.19.59.195
44444/ucp                   DENY  IN    Anywhere
80/tcp                       ALLOW IN    Anywhere
443/tcp                      ALLOW IN    Anywhere
```

192.19.59.190

```
To                           Action      From___________
22/tcp                       ALLOW IN    Anywhere
40811/tcp                   ALLOW IN    192.19.59.190
```

```
44444/ucp                   ALLOW IN    10.19.59.5/24 (v6)
80/tcp                      ALLOW IN    Anywhere
443/tcp                     ALLOW IN    Anywhere
3306/tcp                    ALLOW IN    192.19.59.190
```

In all cases that which is not specifically allowed is denied. This means that an incoming request will be processed, and if TCP, a reset packet returned with an acknowledgement that the request had been received. This is in contrast with what is happening on port 514 in the first firewall, which is the syslog data transfer port. Any requests from outside the cloud will be dropped (completely ignored and no reset packet sent). This is to protect the integrity of the log data as well as lessen the effects of a denial of service attack against that port. Also, in the first firewall, port 22 secure shell is allowed in except for network 172.19.0.0/16, which has launched numerous attacks in the past. Also note that secure shell traffic that gets in via the first firewall is passed though the other two firewalls.

In the case of moving data to and from the data store which is linked to port 40811, direct access is not allowed from outside the cloud and only from the specific data store assigned to the zone (192.19.59.195) and the VM (192.19.59.190) respectively. In evaluating the access allowed to port 44444 (UDP test port) no access is allowed from the cloud or zone firewalls, but a private research and development host, 10.19.59.5/24, is the only host within the zone granted access.

Last, but most important is the access afforded to ports 80 (http, often given the canonical name WWW) and 443 (https, secure port for credit card transactions). These are the heart of an e-commerce system and it is critical that potential customers have access to them. As one can see they are passed across all firewalls and they are hardened on the application layers by settings within the apache web server and the associated data stores. This is an important topic that warrants in depth coverage and will be addressed in part in the next section of the paper.

HTTP Security Layering

Although the basic firewall logic described in the previous section allowed port 80 access from anywhere, it is possible to monitor the traffic flow and identify potential dangerous addresses. Given the packet below from the VM webhost "wwwhost" the destination IP address 192.17.59.179 was blocked because that address kept flooding the service at port 80 with SYN packets and was attempting a denial of service attack.

```
Jun 12 09:45:47 wwwhost kernel: [11806692.157123] [UFW BLOCK] IN=eth0 OUT=
MAC=00:50:56:8c:52:4e:00:50:56:8c:05:a6:08:00 SRC=192.17.59.24 DST=192.17.59.179
LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=53631 DF PROTO=TCP SPT=40173 DPT=80 WINDOW=14600
RES=0x00 SYN URGP=0
```

To accomplish this, a script monitoring the traffic flow identified that the packet inter-arrival rate was much higher than its confidence interval (the average + or – one standard deviation) which in this case was (-1.1 in effect 0) to 4.7 milliseconds and then inserted the deny statement below into the firewall which effectively denies the whole 192.17.59.0 network.

```
Host                Loss%   Snt   Last   Avg  Best  Wrst StDev
192.17.59.179        0.0%    18    0.9   1.8   0.9  13.5   2.9

80/tcp         DENY  IN    192.17.59.0/24
```

There are also easy to implement advantages of using a cloud architecture. For example, the VMs (virtual machines) used to host the http traffic used in e-commerce make it easy to support multiple instances of the apache web server. Typically, SMP (symmetric multi-processing) is available on the operating system level and supported by the apache version two software. The output below indicates that the root level process has allowed five light weight process to fork from it. This five layer approach has several advantages. First, it speeds up the processing because multiple processors can be used. Two, if a denial of service attack occurs it spreads out the effects and lessens the impact on any on process. Last, it isolates the code because each session is its own separate process.

```
ps -aux | grep apach

root     13360  0.0  1.1 230392 11648 ?        SNs  Mar25   0:44 /usr/sbin/apache2 -k start
www-data 14962  0.0  0.6 230416  6232 ?        SN   Mar31   0:00 /usr/sbin/apache2 -k start
www-data 14963  0.0  0.6 230816  7080 ?        SN   Mar31   0:00 /usr/sbin/apache2 -k start
www-data 14964  0.0  0.6 230816  7080 ?        SN   Mar31   0:00 /usr/sbin/apache2 -k start
www-data 14965  0.0  0.6 230816  7080 ?        SN   Mar31   0:00 /usr/sbin/apache2 -k start
www-data 14966  0.0  0.6 230416  6232 ?        SN   Mar31   0:00 /usr/sbin/apache2 -k start
```

In addition there are security mechanisms that can be invoked on the service level, in this case http. The default setting as shown below are designed for general static web pages in a non-production environment. Running e-commerce on a web service changes the environment and makes it critical that the service is reliable and cannot be tampered with. By carefully selecting the directive settings another effective layer of security can be created. For, example the AllowOverride setting below can be very dangerous. By setting AllowOverride to none will stop hacker's from creating an .htaccess file which can override existing security features.

```
<Directory /var/www>
Options Indexes Includes FollowSymLinks MultiViews
AllowOverride AuthConfig
Order allow,deny
Allow from all
</Directory>
```

By setting AllowOverride to none, that will stop hackers from creating an .htaccess file which can override defined security features. So in effect, this feature mitigates the danger from .htaccess files in all directories unless the directory is specifically enabled. However, access can be granted as needed to specific directories such as the /usr/users/*/www directory below. These protections of course are in addition to whatever protection level that might be defined on the file system, directory or file levels. So in effect a security layer is added on the service level.

```
<Directory /var/www>
AllowOverride None
</Directory>
<Directory />
Order Deny,Allow
Deny from all
</Directory>
<Directory /usr/users/*/www>
Order Deny,Allow
Allow from all
</Directory>
```

CONCLUSIONS

Most installations adopting cloud architecture can be expected to have general policies in place concerning firewall logic. However, it is important to understand that if a hacker penetrates the cloud and security layering is not utilized they may gain easy access to the resources of the cloud. To combat this problem a cloud test-bed was configured that contained three layers: the cloud, zone and VM. Traffic that was considered dangerous could be filtered out on the cloud level and VMs within the cloud that only needed to communicate with one another could filter out traffic from other hosts within the zone. In cases where research and development traffic was needed within the cloud, traffic could be forced to follow only a single path within the cloud.

It is also possible to use this strategy to deal with the dynamic nature of computer systems. Specifically, when attack patterns from a given network or site are detected then the firewall(s) can be reprogrammed to block those attacks. In fact, this process can even be automated through scripting logic. In sum, the crucial point is that layering provides multiple chances to stop an attack. Layering is analogous to having three locks on a door, because if one fails hopefully the other two won't. This logic protects the enterprise in general, but also benefits common important services such as HTTP, which may be supporting e-commerce. This layered logic increases the chances that continuous dangerous traffic will be filtered out by at least one of the layers. Further, it often allows for the dynamic application of additional resources so the HTTP service runs faster and is less affected by denial of service attacks.

The primary goal of this paper was to explain the basic design of segmenting a cloud and illustrate via a test-bed how this layering security logic might be implemented. To truly implement such logic would require a global business to first devise comprehensive security policy and then enhance it to take advantage of the layering a cloud could provide. This is not a trivial task and would require a substantial commitment in terms of time and cost. However, if successfully implemented it would more than likely pay off in the long run.

REFERENCES

1. Anthes, G. (Nov. 2010). Security in the cloud. Communications of the ACM , vol. 53 Issue 11, pp.16-18.
2. Arias-Cabarcos, P., Almenarez-Mendoza, F., Marin-Lopez, A., Diaz-Sanchez, D., & Sanchez-Guerrero, R. (2012). A Metric-Based Approach to Assess Risk for 'On Cloud' Federated Identity Management. *Journal of Network And Systems Management*, (4), 513.
3. Basak, D., Toshniwal, R., Maskalik, S. (et.al). (Dec 2010). Virtualizing networking and security in the cloud. *ACM SIGOPS Operating Systems Review, vol. 44 Issue 4, pp. 86-94.* Retrieved from http://bit.ly/1fj5ICO
4. Bohli, J., Gruschka, N., Jensen, M., (et.al) (July-Aug 2013). Security and Privacy-Enhancing Multicloud Architectures. *IEEE Transactions on Dependable and Secure Computing, vol. 10, no. 4, pp. 212-224.* Retrieved from http://www.computer.org.ezproxy.umuc.edu/csdl/trans/tq/2013/04/ttq2013040212.html
5. Caron, E., Le, A., Lefray,A., Toinard,C. Definition of Security Metrics for the Cloud Computing and Security-Aware Virtual Machine Placement Algorithms. *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, pp.125-131.* Retrieved from http://www.computer.org.ezproxy.umuc.edu/csdl/proceedings/cyberc/2013/5106/00/5106a125.pdf
6. Farrell, R. (2010). Securing the Cloud-Governance, Risk, and Compliance Issues Reign Supreme. *Information Security Journal: A Global Perspective*, *19*(6), 310-319. doi:10.1080/19393555.2010.514655 Retrieved from http://eds.a.ebscohost.com/eds/pdfviewer/pdfviewer?sid=bdefb419-29db-4671-8774-08b5eac54e21%40sessionmgr4005&vid=2&hid=4208
7. FORD, S. (2010). Managing Your Global Business With Cloud Technology. *Financial Executive*, *26*(8), 56-59. Retrieved from http://eds.a.ebscohost.com/eds/pdfviewer/pdfviewer?sid=e0f4d132-aab4-4a50-a119-9c03cdc21c0d%40sessionmgr4001&vid=6&hid=4208

8. Géczy, P., Izumi, N., & Hasida, K. (2012). CLOUDSOURCING: MANAGING CLOUD ADOPTION. *Global Journal Of Business Research (GJBR)*, *6*(2), 57-70.
9. Green, M. (Jan-Feb 2013). The Threat in the Cloud. *IEEE Security & Privacy, vol. 11, no. 1, pp. 86-89*. Retrieved from http://www.computer.org.ezproxy.umuc.edu/csdl/mags/sp/2013/01/msp2013010086.html
10. Guster, D. C., Lee, O. F., & Rogers, D. C. (2011). Pitfalls of devising a security policy in virtualized hosts. *Journal of Information Security Research, 2*(2), 75-83.
11. Guster, D., Smith, M., Lebintritt, L. (2014). Using Common Linux Commands to Trace the Origins of Potentially Rogue Processes within a Linux Host (Virtual Machine). *Proceedings from the 2014 Midwest Instructional Computing Symposium, Verona, WI.*
12. Hyman, P. (June 2013). Augmented-reality glasses bring cloud security into sharp focus. *Communications of the ACM, vol. 56 Issue 6, pp. 18-20.* Retrieved from http://bit.ly/MGQGyW
13. Jules, A., Oprea, A. (Feb 2013). New approaches to security and availability for cloud data. *Communications of the ACM, vol. 56, issue. 2, pp. 64-73.*
14. Kaufman, L. (July-Aug. 2010). Can Public-Cloud Security Meet Its Unique Challenges?. *IEEE Security & Privacy*, vol. 8, no. 4, pp. 55-57. Retrieved from http://doi.ieeecomputersociety.org.ezproxy.umuc.edu/10.1109/MSP.2010.120
15. Lesk, M. (May-June 2012). The Clouds Roll By. *IEEE Security & Privacy, vol. 10, no. 3, pp. 84-87.* Retrieved from http://www.computer.org.ezproxy.umuc.edu/csdl/mags/sp/2012/03/msp2012030084.html
16. Mell, P. (July-Aug 2012). What's Special about Cloud Security?. *IT Professional, vol. 14, no. 4, pp. 6-8.* Retrieved from http://www.computer.org.ezproxy.umuc.edu/csdl/mags/it/2012/04/mit2012040006.html
17. Roberts, J., Al-Hamdani, W. (2011) Proceedings from the 2011 Information Security Curriculum Development Conference. *Who can you trust in the cloud?: a review of security issues within cloud computing, pp 15-19.* Retrieved from http://bit.ly/1bfH3jn
18. Seo, S., Nabeel, M., Ding, X. (et.al). (Aug 2013). An Efficient Certificateless Encryption for Secure Data Sharing in Public Clouds. *IEEE Transactions on Knowledge and Data Engineering*, *05 Aug. 2013. IEEE computer Society Digital Library*. Retrieved from http://www.computer.org.ezproxy.umuc.edu/csdl/trans/tk/preprint/06574849.pdf
19. Sun, D., Chang, G., Sun, L., Wang, X. (2011). Surveying and Analyzing Security, Privacy and Trust Issues in Cloud Computing Environments. *Procedia Engineering, vol. 15, pp. 2852-2856.*
20. Takabi, H., Joshi, J., Ahn, G. (Nov-Dec 2010). Security and Privacy Challenges in Cloud Computing Enviroments. *IEEE Security & Privacy, vol. 8, no. 6, pp. 24-31*. Retrieved from http://www.computer.org.ezproxy.umuc.edu/csdl/mags/sp/2010/06/msp2010060024.html
21. Viega, J. (July-Aug. 2012). Cloud Security: Not a Problem. *IEEE Security & Privacy, vol. 10, no. 4, pg.3.* Retrieved from http://doi.ieeecomputersociety.org.ezproxy.umuc.edu/10.1109/MSP.2012.93
22. Zargari, S., Smith, A. (2013). Policing as a Service in the Cloud. *Proceedings from 2013 Fourth International Conference on Emerging Intelligent Data and Web Technologies.* Retrieved from http://www.computer.org.ezproxy.umuc.edu/csdl/proceedings/eidwt/2013/2141/00/5044a589.pdf
23. Zissis, D., Lekkas, D. (March 2012). Addressing cloud computing security issues. *Future Generation Computer Systems, vol. 28, issue 3, pp. 583-592.* Retrieved from http://www.sciencedirect.com.ezproxy.umuc.edu/science/article/pii/S0167739X10002554#