Intrusion Detection and Countermeasure of Virtual Cloud Systems - State of the Art and Current Challenges

Andrew Carlin, Mohammad Hammoudeh
School of Computing, Mathematics & Digital Technology
Manchester Metropolitan University
Manchester, UK

Omar Aldabbas
Al-Balqa Applied University
Faculty of Engineering
Salt, Jordan

***Abstract*—Clouds are distributed Internet-based platforms that provide highly resilient and scalable environments to be used by enterprises in a multitude of ways. Cloud computing offers enterprises technology innovation that business leaders and IT infrastructure managers can choose to apply based on how and to what extent it helps them fulfil their business requirements. It is crucial that all technical consultants have a rigorous understanding of the ramifications of cloud computing as its influence is likely to spread the complete IT landscape. Security is one of the major concerns that is of practical interest to decision makers when they are making critical strategic operational decisions. Distributed Denial of Service (DDoS) attacks are becoming more frequent and effective over the past few years, since the widely publicised DDoS attacks on the financial services industry that came to light in September and October 2012 and resurfaced in the past two years. In this paper, we introduce advanced cloud security technologies and practices as a series of concepts and technology architectures, from an industry-centric point of view. This is followed by classification of intrusion detection and prevention mechanisms that can be part of an overall strategy to help understand identify and mitigate potential DDoS attacks on business networks. The paper establishes solid coverage of security issues related to DDoS and virtualisation with a focus on structure, clarity, and well-defined blocks for mainstream cloud computing security solutions and platforms. In doing so, we aim to provide industry technologists, who may not be necessarily cloud or security experts, with an effective tool to help them understand the security implications associated with cloud adoption in their transition towards more knowledge-based systems.**

Keywords*—Cloud Computing Security; Distributed Denial of Service; Intrusion Detection; Intrusion Prevention; Virtualisation

I. INTRODUCTION

Cloud computing is a growing facet of the technical infrastructure of modern information systems. It provides a way to deliver the demand of users for near consistent access to their data and software resources regardless of their physical position [1]. Many industries have already adopted cloud computing given the benefits, such as the elasticity, agility, adaptability and availability, that it brings to the Information Technology (IT) infrastructure [2]. The cloud model reduces industrial costs by simplifying the process of installing hardware and software updates and ensuring availability and adaptability of computing resources as required. These properties allow resources to be deployed as necessary to manage peak capacity or to support prolonged industrial growth. Cloud computing allows a rapid response to these demands when compared to traditional IT models, where resources has to be managed and installed on-premise causing both high start-up and maintenance costs. In the cloud model, resources are rented, often autonomously, as required from the Cloud Service Provider (CSP). Equally, resources can be returned to the 'pool' when not being used leading to greater resource utilisation efficiency, lower cost and 'greener' computing [3]. This flexible infrastructure allows industries to focus on their own business processes, while the computing elements are managed by a CSP rather than by the company's own IT departments. Some companies choose to make use of internal private clouds allowing them to dynamically deploy their own computing resources as appropriate.

The cloud computing model relies on maintaining a certain level of trust between clients and providers to ensure that client's data is secure and that an agreed Quality of Service (QoS) is provided at all times. There is also trust from the providers to the clients to avoid malicious activity against the provider either through direct, e.g., insider attacks, or indirect means, e.g., security flaws in applications that are deployed on the cloud. Changes in the user's use of resources is monitored by the CSP. Unusual discrepancies in this use can affect trust in the user and therefore affect the services they receive [4].

One issue that has hampered the uptake of cloud computing by businesses is the issue of security. This covers the security concerns of all stakeholders from end users, to clients and to the CSPs themselves. The relationships between clients and CSPs are underlined by service agreements, which define the service that clients should receive. These agreements define the responsibilities of all parties with regards to accessibility, data integrity, confidentiality and security.

The power of the cloud is a tempting target for exploitation from attackers aiming to launch further attacks. In 2011, a hacker used Amazon's Elastic Computing Cloud Service (EC2) to attack Sony's online entertainment systems, compromising more than 100 million customer accounts. This was the largest data breach in U.S. history [5]. A similar attack was later used to prevent users from logging on to Sony services [6]. Another example of the power of the cloud being utilised for malicious activity is shown in the work by Thomas

Roth (2011) [7]. Roth has created a program that runs on Amazon's EC2 to brute-force wireless network passwords by testing approximately 400,000 passwords per second. According to his research, 'the average password is guessed in six minutes' at a cloud computing cost of $0.28 to $1.68 per minute. Previously, it would have been extortionately expensive to run such an attack, but based on the aforementioned examples, cloud computing makes these costs negligible [7]. These examples, demonstrate the potential of the computing power available as a potential launch pad for further attacks.

Similarly, the large volumes of data stored in the cloud make it a highly attractive ultimate target to attackers [4]. There are many references in the literature [4, 8] relating to the increasing number of globally reported cyber-attacks that aims at stealing businesses data. These come in the form of the increasing number of targets who have experienced difficulties in accessing data, suffered from identity fraud or been the victim of phishing scams. There have also been a number of high profile attacks targeting large industries, which can have detrimental effects even if they are dealt with quickly and efficiently [9]. These effects include loss of client confidence, misuse of company resources and loss of revenue. For instance, the largest DDoS attack in Norwegian history disrupted the websites and online payment systems of five banks, three airlines, two telecommunication firms and one insurance company. This attack was committed by an individual highlighting the computing power that can be leveraged by a single source [6].

Security Researchers Mary Landesman and Dave Monnier [10] have reported a 'meaningful increase' in attacks on cloud hosting providers [insert ref here]. These attacks commonly use set-up or hacked accounts to deploy command and control servers to conduct malicious activities. It is reported that 47% of phishing attacks were from exploited web hosts. This is because by updating the configuration of a single web host hundreds or even thousands of websites can be infected with phishing pages. Known attacks have used such approaches to compromise nearly 20,000 websites. This demonstrates that attackers are exploiting cloud computing to access the computing power required for larger scale attacks [10].

It has also been seen that attacks against the routers that control traffic and provide the backbone of the internet are growing in line with other cybersecurity issues. DDoS attacks, such as those against Cloudflare and Spamhaus, are increasingly abusing the Simple Network Management Protocol (SNMP). Researchers have found that in the month of May 2014, fourteen separate DDoS attacks made use of SNMP amplified reflection attacks [11]. This emphasises the rate at which attacks are evolving and highlights the importance of constantly evolving defence systems. Issues are further complicated by the various deployment models of the cloud and the responsibility of the various parties for security in each of these setups [12].

This paper aims to investigate the latest defensive systems proposed for use against DDoS attacks targeting the cloud model. In Sections I and II, the key areas of virtualisation and intrusion detection and the relevant security issues with each are examined. Section III presents a classification of intrusion detection in the cloud and highlights the main challenges facing their deployment. Section IV explains how countermeasures proposed for traditional networks are ineffective in cloud environments. Section V present the latest developments in the areas Virtual Machin (VM) security. Section VI, presents intrusion detection and prevention systems in cloud systems. Section VII focusses on defence systems against DDoS in the cloud. The security issues across each of these areas discussed in Sections V to VI, are investigated along with proposed solutions. A summary of the proposed solutions across these areas is presented in Section VII. Section VIII concludes the paper and highlights future research avenues.

II. Overview of Cloud Technologies

A. Virtualisation

Virtualisation is the key concept behind the cloud computing model. It allows programs to be portable across platforms as well as regulates their scalability, monitoring and security [13]. This technology provides interfaces, allowing for virtual process or system machines to be mapped onto the underlying hardware. Such interfaces allow guest instructions provided by the user of the Virtual Machine (VM) to be converted into host instructions through Dynamic Binary Translation. Instructions are converted in blocks rather than individually to provide greater efficiency and allow them to be saved for reuse in software caches [14].

A key benefit of virtualisation is that it allows multiple users to co-habit a single physical machine. This leads to resource consolidation, increased capacity, mobility and makes the system easier to maintain. Virtualisation efficiently supports many tenants, while attempting to isolate them from each other. It provides load balancing through dynamic provisioning and allows the migration of VM's between physical resources. Simultaneously, virtualisation 'poses a major security risk' given the difficulty in ensuring that different instances running on the same physical machine are fully isolated [15]. Vulnerabilities in the VM or VM Manager (VMM) can be exploited to bypass security restrictions or to gain unauthorised privileges.

Loganayagi et al. [16] argues that securing virtualisation technologies will improve cloud security. The key security mechanism of VM is isolation. Isolation allows multiple users to co-habit the same physical host without data leakage occurring across users and without unauthorised users being given access to the VMs of others. However, the scalability features of VMs can still allow some issues to be exploited, e.g., expose memory and process management functionalities leading to privilege escalation attacks. Defence approaches involving isolation can be split into those that isolate the running of VMs and those that focus on the isolation of shared resources [17].

B. Cloud Architecture and Service Deployment Models

According to Jang-Jaccard [4], the typical architecture of a cloud computing environment can be divided into 4 layers (see Figure 1):

- ***The hardware layer:*** This is usually a data centre with responsibility for the physical cloud resources including, servers, routers, switches, power, cooling systems, etc.
- ***The infrastructure layer:*** This layer is also known as the virtualisation layer and is used to pool resources by partitioning the physical resources. This is an essential component of the cloud computing architecture, as dynamic resource assignment and other key features are only made available through virtualisation.
- ***The platform layer:*** This layer consists of operating system and application frameworks, reducing the load of deploying applications directly into virtualisation containers.
- ***The application layer:*** This is the domain of applications, which can make use of the automatic scaling features of the cloud.

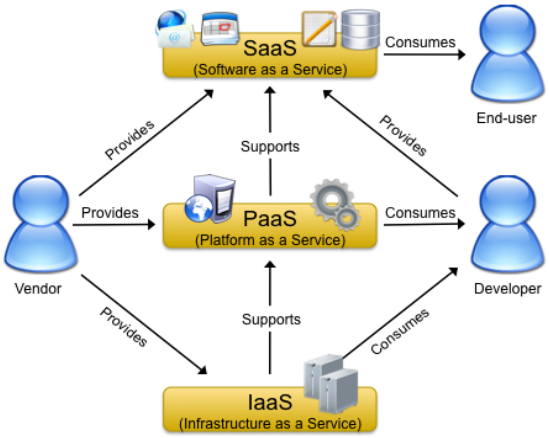

Fig. 1. Service deployment models of cloud computing [18]

There are many different deployment models offered by cloud providers to adapt to client requirements. These are typically divided into three groups [15]:

1) Infrastructure as a Service (IaaS) provides the client with access to hardware in a 'rented' capacity. The space on hardware dedicated to a particular client will vary depending on their demand, allowing the hardware to deal with fluctuations and spikes in requirements. Out of the box, IaaS provides only a basic level of security, e.g., perimeter firewall, and therefore applications deployed in this manner will need higher levels of security provided at the host. The responsibilities in this model vary between providers with some taking responsibility for the security of the infrastructure including the hypervisor and below, with the rest being left to the client.

2) Platform as a Service (PaaS) allows users access to virtual operating systems on top of the hardware layer. In such a setup, the CSP would typically be responsible for the security of the hardware and operating system elements. The client is responsible for the security of any further software that would be installed on top of these elements, i.e., at the application layer, the responsibility for security lies with the client. The provider may use metrics to measure the security of the applications deployed on their services. Below that in the hierarchy, it is up to the provider to offer strong assurances that data will not be accessible to other applications.

3) Software as a Service (SaaS) adds a further layer on top of the PaaS, with software being provided on top of the virtual operating systems. The security of the software is a priority as they provide a gateway to further down into the cloud structure. In this model, the client is dependent on the CSP for security measures, because company data is stored at their data farms. This storage could be located or replicated to aid availability to anywhere in the world.

Across all models, the security complexities are amplified due to the composite relations between the different deployment levels and user requirements. Security responsibilities needs to be particularly considered when using IaaS and PaaS for the development of other IT products to avoid introducing further security issues. Often, all deployment models and their specific responsibilities will be outlined in the QoS contracts drawn up between all relevant parties.

C. Cloud Vulnerabilities

The openness, elasticity, and amount of data stored in clouds make them attractive targets for attackers. The expansiveness of cloud operation across geographical and technological regions also brings with it the security issues associated with each of these areas. Traditionally, networks were less distributed, making defence mechanisms focused on insider attacks. Given that networks are now far more de-centralised and are connected globally via the Internet, the security risks in these systems have grown exponentially. Attackers are now able to target globally without concern over geographical location. Scripts and tools for launching attacks against networked systems are readily available on the Internet and require minimal user skills to execute.

IT users want to be assured that their data is secure. In the cloud computing paradigm, security responsibilities are passed to an outside agency. This can leave users feeling vulnerable, because they no longer control the physical storage where their data is residing and the legal frameworks around its protection. There is also the fear of limited availability or the introduction of further vulnerabilities, e.g., SQL injection and buffer overflow, which can be exploited through web browsers. Moreover, user interactions with the cloud are governed by traditional Internet protocols, e.g., HTTP, which makes it more difficult to identify attackers and easier for attackers to implement distributed attacks [19].

This paper pays particular attention to the vast increase in the number of DDoS as detailed by Wang et al. [8]. DDoS is an attack on system's availability to serve legitimate users. These attacks can be constructed in a number of ways. Table I summarises the main attacks aimed at causing this kind of disruption.

TABLE I. COMMON DDOS ATTACKS

Type of Attack	How it works
Flooding	Flooding can occur through all network and application layer protocols, e.g., HTTP, TCP, UDP, ICMP, etc. It attempts to saturate the network bandwidth by sending a large volume of packets from single or distributed sources so that it is unavailable to process legitimate user traffic. Flooding can be direct attack against the network or application, or reflective attacks via zombie machines.
Spoofing	This approach is used to falsify the origin of a network traffic to bypass filters, hide the source of an attack or gain access to restricted resources or services.
User to root	Aims to gain administrator (root) access privileges for a non-authorised account.
Port Scanning	Provides a list of open ports and the services provided by each; these can then be targeted by other attack methods. Port scanning is used in the first stages of an attack cycle and comes in many forms such as TCP SYN, TCP ACK, TCP ECHO, TCMP SWEEP, etc.
Oversized XML	The attacker sends a very large XML document (several megabytes in size) that contains elements, attributes or namespaces with large names or content. The Document Object Model parses documents into memory in their entirety to be analysed increasing memory requirements by a factor of 2-30.
Coercive Parsing	The attacker sends malformed XML aimed at clogging up CPU cycles by incorporating many namespace declarations or by simply using very deeply nested XML structures.
Web Service-addressing Spoofing	This is an extension of the spoofing attack, where the ReplyTo or FaultTo address in a SOAP header is falsified leading to a reflective attack.
Reflective attack	Request messages are sent to reflector machines via zombie machines containing the spoofed source IP address of the victim. The genuine replies to these requests are then sent to the victim causing flooding. Such attacks include ICMP ECHO reply flood, Smurf attack, Fraggle attack, DNS flood and SYN ACK(RST) flood.

In traditional networked systems, the disruptions caused by a DDoS attack against a particular target is limited to that target's resources or services. When this paradigm is transferred to the cloud, the potential for disruption crosses traditional organisational boundaries. This is because data is managed by a common CSP and the data of different organisations may be stored on the same physical hardware. Therefore, the scalability of cloud computing is what presents its main security challenges when compared to traditional networks.

III. INTRUSION DETECTION TECHNIQUES IN CLOUD

A. Intrusion Detection

Intrusion detection is the first step in identifying a malicious behaviour against a system. The key challenge is to reliably differentiate between legitimate users and attack traffic. There are two standard approaches used by Intrusion Detection Systems (IDS), these are knowledge-based intrusion detection and behaviour-based intrusion detection [20].

Knowledge-based systems must possess an attack description, typically a signature that can be matched to attack manifestations. Signatures range from simple pattern matching to network packets, such as those used in BRO, ASAX and NADIR systems [21], building up to neural networks that map multiple sensor outputs to abstract attack representations. Jang-Jaccard [4] argues that signature-based systems are ineffective given the constantly evolving landscape of cyber-attacks. While it is true that this approach requires constant updates as new signatures are identified, the simplicity of its structure allows it to be rapidly deployed across systems. In addition, the knowledge databases used provide effective attack classification allowing a more directed response to be triggered.

Behaviour-based, also known as anomaly-based, systems are designed to evolve to meet new, previously unseen threats. They involve monitoring network attributes and assume that the behaviour of malicious parties is noticeably different to that of legitimate users. This assumption allows anomalies to be flagged and alarms to be raised. These systems often require a training period to build a model of network attributes, which raises the cost of their implementation. The usability of these systems is dependent on the False Alarm Rate (FAR) that they generate, which is made up of both false-positive alarms (raising an alarm for legitimate traffic) and false-negatives (the failure to register an intrusion attacks). This approach can struggle to classify attacks allowing only a general system response to be triggered, yet, it can respond to previously unseen cyber-attacks.

Compared to IDS, Intrusion Detection and Prevention Systems (IDPS) include preventative measures to stop attacks in real time rather than simply detecting them once they occur. These systems follow the design principles of IDS, but also take preventative action such as logging a user off, initiating system shutdown, halting the system or disabling connections [22].

B. IDSs Classification and Challenges

In general, IDS systems designed for cloud computing can be classified into four main categories [23]:

1) Host-based IDS (HIDS): These systems monitor and analyse log files, security access and user login information to detect intrusive behaviour.

2) Network-based IDS (NIDS): These systems monitor IP and transport layer headers with behaviour being compared with previously observed behaviour in real time. This approach does not work with encrypted network traffic.

3) Hypervisor-based IDS (HyIDS): These systems allow users to monitor and analyse communication between VMs, within the VMM-based virtual network and between the VMM and VMs. These systems benefit from an availability of information that can be analysed to detect intrusions.

4) Distributed IDS (DIDS): DIDS consists of a number of IDSs (HIDS and NIDS) placed across a large network. These individual IDSs communicate with each other via a central analyser, which aggregates system information from the different IDSs. This system benefits from the qualities of both HIDS and NIDS to detect known and unknown attacks. However, there is a high computational cost in the communication between these systems. In a cloud computing

environment, the central analyser can be placed on a host machine or at the processing server. However, if the analyser is compromised or unable to communicate, then the system will not be able to react to further threats.

IDPS can prove to be an invaluable tool in the early detection of malicious activity helping to prevent attacks from succeeding. They can also gather forensic evidence. However, traditional IDPSs are largely inefficient when applied to cloud computing given its openness. Patel [12] investigates the requirements of IDPS in the cloud architecture given the ineffectiveness of traditional methods by asking what criteria and requirements should an IDPS meet to be deployed on the cloud? Which methods or techniques can satisfy these requirements? The list below outlines some of the challenges that traditional IDPS struggle to counter:

- They do not scale to deal with cloud requirements and do not satisfy the requirements of high-speed networks.
- The traffic profiles of networks changes frequently rendering the audit data used to train the IDPS unsuitable very quickly.
- They generate high false alarm rate [24].
- There is no uniform standard or metric for evaluating an IDPS, which can often lead to misleading information as to their effectiveness.
- It is very difficult to identify internal intrusion attacks given that correctly configuring the systems and implementing organisational policies is a difficult task.

IV. Distributed Denial of Service (DDoS) Attacks and Traditional Countermeasures

The most common attack vector that has been used to attempt to adversely affect cloud services is DDoS attacks. A DDoS attack aims to render the computing resources of the victim unavailable by modifying the system configuration or by sending it too high workload. Moore [25] determined that in 2006 the average rate during a DDoS attack was 500 requests per second and that attack typically lasts less than five minutes. These figures may well now be considered out of date with Goel et al. [26] noting attacks with a rate of greater than 100Gbs.

DDoS attacks can be divided into two general categories, application level attacks and infrastructure level attacks. In application level attacks, e.g., HTTP flood, zombie machines establish TCP connections to the victim server and send legitimate requests. Systems used to detect such attacks can struggle to differentiate between attacks and busy periods, such as the start of the workday where many legitimate users may attempt to access resources simultaneously. One method for dealing with these attacks is the use of CAPTCHA puzzles. However, these puzzles are only suitable for an initial user login or registration; the user may become frustrated with the service if they are used any more frequently.

Infrastructure level attacks require the attacker to send a flood of packets to the victim server in to saturate or bottleneck the victim so that it can not respond to legitimate requests. For this type of attack only the victim's IP address needs to be known. Typical direct infrastructure layer attacks include TCP flood, UDP flood, ICMP flood and SYN flood.

Traditionally, well-known countermeasures have focussed on dealing with DDoS attacks through a variety of methods devised around the questions [27]: (1) Where is the attack detected? (2) How is the attack detected? (3) What is the response mechanism? (4) Where to apply the response mechanism? (5) Where is the control (decision) centre from which filtering rules are taken?

The most common DDoS defence approaches combine elements located in the source-end and victim-end in to combine their advantages. However, the use of multiple components leads to gaps in coverage, which can be exploited. The source-end is the location from which the attack is launched; this is the best place to intercept an attack as it causes the least disruption to legitimate traffic. However, distinguishing between legitimate and malicious traffic at this point is a serious challenge. D-WARD [27] is a system that employs a firewall at the source end. It gathers 2-way statistics from the border routers. This introduces significant overhead because D-WARD is continuously monitoring and classifying traffic based on IP address, comparing statistics and applying filtering rules. The operation of D-WARD affects the speed of the entire network whether there is an attack or not. Beitollahi et al [27] suggest that there is no benefit for deploying source-end firewalls considering the overhead and performance loss they introduce. Yet, a user would not want his networks to be compromised and turned into a pad to launch further attacks. This is more critical when considered in the cloud computing paradigm, where undetected intrusion has the possibility of giving an attacker access to a far greater amount of resources than a traditional network could provide. The CSP would need to balance the the threat of becoming a source of an attack with the detriment in service provided to legitimate users.

DDoS traffic is easier to identify at victim-end points. IDPS at these points are effective at generating attack signatures, which they can then be used by upstream routers to rate-limit or filter traffic. However, by this point the bandwidth of the network could be saturated. Moreover, these infrastructural approaches require the cooperation of multiple Internet Service Providers (ISP) to cover the required range of administrative domains. There are also issues with security and authenticating the communication channels.

For the reasons given above, the majority of traditional DDoS countermeasures are ineffective against application layer attacks. This is mainly because the packets have been transferred and the TCP handshakes have been completed meaning that the packets appear to be legitimate. Packet sniffing protocols are therefore ineffective at this level.

V. VM Common Security Vulnerabilities and Defence Mechanisms

Virtualisation is the key underlying technology of the cloud computing model and therefore its security needs to be considered as the foundations of any proposed system. Particularly, the fundamental weaknesses in the VM architecture need to be addressed to enhance security across other layers. For instance, attackers on the same physical

machine can use malicious code to get control of other VMs. They can then deploy a class of rootkits, e.g., UMBR, which operate under the operating system. These rootkits cannot be removed easily and are difficult to detect [13]. There are also attacks targeting VM migration.

Isolation is the key security feature to protect VMs from malicious attacks. The isolation-based defence approaches can be split into those that isolate the running of VMs and those that focus on the isolation of shared resources [17]. The first approach can limit the ability of the system to schedule the work of legitimate VMs. To implement the second approach, a monitoring and mediation mechanism is needed to probe all resource requests and to allocate these requests to VMs. In addition, to implement the strict policies needed to enforce isolation takes many OS hooks and is difficult to enforce across a large-scale distributed system.

Volokyta [28] suggests a VM Monitor to secure VMs. The proposed system intercepts system calls and maintains a log file of system warnings. The authors do not give the practical design details and experimental results. Therefore, it is not possible to comment on the performance of this system. However, VM management systems are frequently part of security design specifications.

Yu et al. [17] consider Chinese Wall properties, where an object can be read if the subject has accessed a prior object from the same dataset or the objects conflict of interest is set to new. This system records the behaviour of VMs to obtain traces used to calculate the Aggressive Conflict of Interest Relation (ACIR) or Aggressive in Ally with Relation (AIAR). Isolation rules are combined with constraint relations to get the access matrix, which records the maps to give dynamic updates between the VMs and hosts. An algorithm is implemented to guarantee isolation between conflicting users. In this approach, specific monitoring systems are not required, but a trade-off is made between security and resource utilisations. To further improve this system, more efficient methods for conflict analysis are needed. A suggestion for enhanced isolation using ACIR and AIAR to describe constraint relations has been put forward by in [17].

Lui et al. [29] suggest a framework for enhancing VMs security in clouds using module measure. Metrics of the executables running in VMs are taken and compared to a reference table of trusted measurements. This aims at combating user-level security in SaaS, where many individual users access a single instance of an application. In this framework, a trusted VM is used to monitor other VM instances, meaning that the status of the measurement module needs to be noted to ensure that the system can be trusted.

Another approach, presented by Williams et al. [30] is to use N-version programming in the construction of VMs. The authors introduce diversity in VM design to avoid a sequence of events that leads to failure. This approach lends itself to automation making it scalable, but can make the development of compatible systems difficult. The proposed structure provides diversity during execution through Address Space Randomisation (ASR). This approach does not remove vulnerabilities, but aims to make them more difficult to exploit, because an attack that works against one binary will not work against another. This means that only a single instance of an application will be affected during each attack.

To recapitulate, virtualisation poses a number of security issues that need to be addressed. A benefit of virtualisation is its ability to allow users to isolate VMs and resources, which, theoretically, enhance security. A number of solutions have been proposed to monitor and enforce the principles of isolation and thus secure the virtualisation layer. A common suggestion is to use a VM as a designated manager to monitor the operation of the other VMs in the network. If only a single management machine is used, then this could create a bottleneck in the system especially in an architecture where additional VMs can be generated and deployed autonomously. The systems reviewed could be made more suitable for a cloud environment if monitoring VMs could also make use of the scalable nature of the cloud. This would mean that they could increase their number autonomously to ensure efficient management of resources and monitoring of isolation throughout periods of operation.

VI. IDPSS IN THE CLOUD

IDS and IDPS face difficulties when transferred from traditional networks to cloud-based designs. Issues such as those with their deployment locations and the separation of legitimate traffic from malicious traffic pose real challenges with their implementation. Defence systems must successfully determine the need for their use before they can be executed, otherwise the user experience will deteriorate. The latest developments in IDPSs are investigated in this section''.

Al-Jarrah et al. [31] suggest embedding the temporal behaviour of attacks into a Time Delay Neural Network (TDNN) model to defend against probe or reconnaissance attacks. The suggested system works on a universal IP plan as the relationships between the inputs are the keys, and no range or class of IP addresses is used. This makes this system suitable for use in cloud computing given the scalability that it offers. However, an autonomous method for including relationships for newly generated nodes would need to be created. The experimental results show the approach to be effective when tested against the DARPA Intrusion Detection Evaluation [32] and other IDS systems such as SNORT. The overheads of the system are not discussed or compared to other techniques, though the authors state that their 'system is characterised by high throughput because after the system is trained, it takes constant time to detect any attack'.

Alqahtani et al. [22] put forward a system to prevent SQL injection in cloud computing web-based systems using signature-based approaches. They focus on the application layer, because web software services contain the majority of security vulnerabilities in cloud systems. Automated tools, such as SQLmap, are identified as having the potential to be used maliciously by hackers to attack cloud-hosted databases.

The presented evaluation method provides suitable metrics for measuring the quality of an IDPS system. These include vulnerability detection, average response time and number of false positives. An improvement of this system is to implement these measurements in other test designs to provide suitable data for comparing IDPS systems.

SNORT and OpenFlow are combined by Xing et al. [33] to produce an IDPS (SnortFlow) that can reconfigure a cloud network system in real time using Iptables. The IDPS is made up of four components:

1) Cloud cluster – this is based on the efficient parallel virtualisation solution, XenServer.

2) Open Flow Switch – this connects resources on different cloud servers.

3) Open vSwitch – this is the software version of the switch and is implemented in one of the domains of the Xen hypervisor.

4) Controller – this provides centralised control over the enabled infrastructure. A POX controller is easy to program and can synchronise both physical and virtual networks.

This approach considers the status of the network when deciding, which actions to take based on the findings of traditional IDS approaches. The optional response is selected using the graphical attack IPS NICE [33]. This decides if a response is necessary and chooses from options including, traffic redirection, traffic isolation, deep packet inspection, MAC address change, IP address change, block port or quarantine. However, every packet is monitored by SNORT, which could create a bottleneck, especially under large-scale DDoS attack conditions. The packet dropping rate of SNORT increases dramatically once the traffic rate exceeds 45,000 packets per second [34] thought it does have a better throughput than some other similar systems such as Suricata [35].

Hassani [36] combine IDS, IPS and hybrid detection techniques (pattern matching and anomaly detection) to address the issues of each individual approach. This effort focuses on distributed attacks coming through the infrastructure layer. Individual system components each have their own weaknesses, which may result in some attacks going undetected if attackers alter the timing used during attacks to make these appear as random individual requests. The collaborative IDS put forward makes use of the entire network to correlate events and to deduce a distributed attack that occurs in several places. There is no implementation of the suggested system, which makes it is difficult to measure the efficiency and cost of this approach.

Most current Intrusion Response Systems (IRS) use static matching to decide a suitable response action to an attack. The issue with this approach is that it does not consider the status of the entire system. Alazab et al. [37] suggest a system to improve detection efficiency. The proposed system uses an Intelligent IDS (IIDS) built up of SIDS, AIDS and IRS. The mechanism links the state of an attack with a response to raise an alarm, or to audit, hold, abort, disconnect or refuse packets. The IRS uses two stages to assess the potential risks of the anomaly using the Microsoft DREAD model shown in Table II.

Initially, the SIDS examines the content of the user request for currently known intrusions. This is followed by the AIDS step to accommodate the shortfalls of the SIDS. The AIDS assumes that any request received from the user is an anomaly unless proven otherwise.

TABLE II. MICROSOFT DREAD MODEL [36]

Damage Potential	How great is the damage if the vulnerability is exploited?
Reproducibility	How easy is it to reproduce an attack?
Exploitability	How easy is it to launch an attack?
Affected Users	As a rough percentage, how many users are affected?
Discoverability	How easy is it to find the vulnerability?

A risk assessment matrix is then used to determine whether the request is fulfilled or which action is taken. The IIPS is flexible enough to accommodate different web application architectures. This is aided by the fact that the communication of the SIDS and AIDS are based on web application architecture.

VMFence is a system put forward by Jin et al. [34], which uses a VM Monitor based IPS to monitor network flow and file integrity in real-time. The defence of the network and file integrity protection varies with the state of the VM. This approach is based on the fact that virtualisation-based cloud computing comes down to the security of virtualisation itself. The system uses a privileged VM and contains 5 phases:

1) Detection – This captures all network packets and dispatches them to other detection processes according to their MAC address.

2) Policy Updating Component – Used for intrusion response and collects all alerts.

3) Front-end to Back-end Communication – Updates the firewall rules in real-time.

4) File Integrity Monitoring – Observes read/write operations.

5) Notification – Receives service type and sensitive files defined by cloud users. This unit also collects alerts for the cloud provider.

Snort is used as an IDS. Iptables are used as a firewall with policies being updated via a shared page located in the XenStore. The back-end and front-end communicate via the event channel. This approach is faster than traditional response by network. File integrity is monitored on the blktap mechanism that can directly manage disk activities with small performance overhead. VMFence uses a privileged VM to monitor the other VM nodes, which cause a bottleneck in the system. However, this system can make use of scalability properties of the cloud to relieve any bottlenecking.

The reviewed solutions in this section outline many issues with the design of current IDS and IDPS systems. Common themes can be drawn across all of the proposed systems regarding the size of system overheads, how to react once an alert is triggered and how to effectively reduce the false alarm rate. Many proposed systems focus on detecting a single style of attack or protecting a single layer of the cloud architecture. Although these approaches can be successful in providing security to one part of the cloud, a cohesive and adaptable system is required to avoid the layering of individual security components. The use of many individual defence systems can lead to the development of further vulnerabilities along the protection vulnerabilities of each of these systems. There is no standardised procedure for measuring the quality and

effectiveness of an IDS making comparisons between proposals difficult. Implementing a standardised set of metrics would enable performance comparison of various systems and allow the strengths and weaknesses of each to be identified as well as highlighting areas for further research in the field.

VII. Systems For Defending Against DDos in the Cloud

DDoS attacks can render CSPs unable to provide their users with the service as outlined in their QoS documents and/or they have their resources manipulated to launch an attack against external targets. This section builds on the knowledge of the cloud architecture and IDPSs to analyse proposed systems aimed at protecting the cloud against DDoS attacks.

Yang et al. [38] propose a trace-back and filter system to protect the cloud from DDoS attacks. The current packet tracing methods of Probabilistic Packet Marking (PPM) and Deterministic Packet Marking (DPM) will become ineffective with the introduction of IPv6. To overcome this, trace-back is implemented by adding a tag to Service Orientated Architectures (SOA) packets to record the route taken. This method is rather limited in its effect to counter DDoS attacks, because the tag is only added to the packet once it is relatively close to the server. When an attack is launched via the Internet, SOA packet tracking does not provide enough information to identify the source of the attack. The tests presented by the authors do not consider spoofed IP addresses or the use of zombie machines in the attack. The attacker does not need to cover the locations of zombie or spoofed machines, as the overall source of the attack will remain protected. Another weakness is assuming that even whilst under attack, the system will operate properly and communicate correctly with the upstream filters. These filters are expected to remove attack traffic, but the authors assumes that attack traffic is all coming from single sources. If the bandwidth is flooded, then the deployment of these resources can not be relied upon.

Another trace-back model is suggested by Joshi et al. [39] using DPM and training data to inform the filters in a neural-network. The future of DPM may be limited with the introduction of IPv6 and the fact that trace-back tags can only be introduced once a packet is within the cloud network. This system has a success rate of correctly identifying approximately 75% of attack traffic, though its ability to detect currently unknown attacks is not researched. It has a significant time variation in the detection rate of attack traffic from 20ms to 1s; an overhead that may cause disruption to legitimate users when accessing systems even if an attack is not taking place.

Karnwal et al. [19] introduce a filtering tree to act as a service broker within a SOA. They investigate the vulnerabilities in standardised cloud APIs and how they can be exploited when used in provisioning, management and monitoring of services. They propose adding a signature reference element to each SOAP request to ensure that it comes from a legitimate source. Double signatures are created using hashed characteristics of each SOAP envelope, such as the number of child or header elements. The client IP address is also maintained in the message header along with a puzzle that is stored as part of the WSDL file. Scanning each packet individually will eventually lead to bottlenecks. IP trace-back is put forward as a method for discovering the source of attacks and updating defence systems to drop packets from that location. However, the system will remain ineffective in dealing with flooding attacks from distributed sources or those using spoofed IP addresses.

Vissers et al. [40] work aims to deal with DDoS attacks at the application layer. Their solution aims to protect primarily against HTTP flooding, Oversized XML, Coercive parsing, Oversized Encryption and Web Service-addressing spoofing. A reverse proxy is added as a filter to intercept all service requests. It is claimed that this filter adds no overhead to the cloud operation and that users experience no effect to their service. The web service itself is only accept requests that come from the defence server. If the server itself is directly flooded, then the server's reaction to legitimate requests will be affected. Initially, the system processes the HTTP header to get the size of the request, to see if the packet is oversized. Reading the header also means that the number of requests from a single client in a given period can be limited to enable fast detection of HTTP floods. To provide further security, strong authentication is imposed on users attempting to connect to web services. This helps protect against 'meek' attacks, where a large number of zombie machines make a low rate of individual requests that can collectively lead to DDoS. The SOAP action header is then taken and used to determine the requested operation of the packet without having to examine all of the XML content. The action header could be spoofed by malicious parties and therefore further checks are required to verify if this is the case. The second phase involves processing the key properties of the XML content and comparing them against the pre-determined attack models. WS-addressing spoofing is also extracted before the header details and content processes are compared to determine if they match. The results presented are encouraging, but are limited in the sense that only a single platform set-up is tested with a single-target web application. In addition, the attack tool could not generate attacks involving encryption or signatures and oversized encryption attacks were sent directly from legitimate sources. This system requires further development to cover multiple platforms and applications to reflect its performance in real world scenarios.

Another application layer DDoS IDPS specifically designed to deal with Low and Slow (LOS) attacks is presented in [41]. These attacks are rarely detected using pattern matching or threshold measuring techniques given their low resource consumption approach. The authors propose a reference-based architecture to mitigate DDoS attacks by utilizing a Software Defined Infrastructure (SDI). Senthilmahesh et al. [42], Mathew et al. [43] and Tang et al. [44] describe techniques used for detecting LOS DDoS in the proposed system. The system introduces a 'healing' approach that is implemented if an intrusion is detected. This approach involves migrating legitimate users from compromised to newly generated VMs. 'Shark Tanks' are introduced as quarantine areas for potentially malicious traffic. This allows suspect users to be monitored more closely, while continuing to receive a suitable level of application access in

case a legitimate user is wrongly redirected. The locations of the Shark Tanks are disguised using OpenFlow switches, which can rewrite packet headers so that attackers are unaware that they are being monitored. The system and clusters are described using Domain Specific Language (DSL) and the use of VMs means that the system is scalable. Two concrete implementations of the design are put forward. However, the results of these implementations are not compared against existing systems or each other.

Wang et al. [8] use the autonomous generation properties of the cloud to base service delivery on randomly generated and assigned proxy nodes. This restricts the information that attackers can determine through reconnaissance attacks with all internal IP addresses being kept hidden. Attacking machines using spoofed IP addresses become ineffective, as they will not receive server reassignment messages. Strong authentication techniques are proposed to prevent external attackers from accessing proxy nodes. The application server will only accept requests from a designated ring of proxy nodes and from clients who have made a successful connection. Connections are monitored by tokens passed between the user and the authentication server. The presented solution focuses on preventing 'insider' attacks by imposing strong authentication to connect to a proxy node, which will stop external attackers from launching an attack unaided. The benefits of the proposed system are that it make use of the properties of the cloud allowing it to scale and it does not require universal deployment to provide protection. However, the system relies entirely on the IP addresses of key components, mainly the application server, being kept hidden but there is no explanation as to how this can be achieved. Currently, all of the datasets used to generate the models used by the system are stored centrally by the application server. The authors acknowledge as being an issue, the application server itself may become even more of a target. An effective fix to this problem is to distribute these datasets.

Wang at al. [8] propose a greedy algorithm to deliver a 'near-optimal' method for assigning users to proxy nodes. Users are 'shuffled' between newly generated nodes when an attack is detected against a proxy node. The repeated shuffling of users allows the system to identify the insiders provoking the attack. This work is extended by Jia et al. [45], who introduce a selection of algorithms to optimise runtime reassignment plans. They optimise the greedy algorithm in the form of a dynamic programming algorithm. The real-world implementation of this algorithm is very computationally expensive. Both solutions use the number of persistent bots, containing the intelligence to follow migrating servers, as a key parameter for calculating the optimised 'shuffling' pattern. However, in the real-world this value is unknown and can only be estimated.

Jia [45] extends the system proposed in [8] to provide security at the application layer and to offer security for anonymous users by removing the need for strong client authentication. This produces a generic DDoS protection product that can be deployed by non-ISP organisations. This product is efficient at mitigating DDoS attacks and is more cost-effective than static based systems. Both [8] and [45] do not discuss how the attack is detected by the proxy node, they only give the response that is used to identify and dispel the source. Implementation overheads are discussed and are dependent on the number of shuffles required. They are also dependent on the size of the geographical area covered, which could be global in a cloud computing context.

Another attempt at using the capabilities of the cloud in a DDoS defence is proposed in [17]. This system aims to protect individual cloud users by creating clones of virtual IPS to filter traffic. A queueing algorithm is defined to determine the number of IPS clones necessary to defeat the DDoS attack and maintain acceptable QoS. This system is based on the assumption that to defeat DDoS attacks, the defence system must have access to greater resources than those of the attackers. Compared to the research in [25], this is feasible when applied by a CSP. This means that DDoS attacks are currently unlikely to be able to affect an entire cloud service. However, the cloud resources available to individual clients are likely to be more limited making them still susceptible to these attacks. The authors assume that the number of service requests follows a Poisson distribution during both normal usage and attack periods, and that the rate of legitimate requests remains constant during both periods. Therefore, the average time that a packet is in the system provides a suitable measure of QoS. The theoretical results provided assume that the IPS cloning solution is effective and that the cloud contains enough idle resources to overcome the attack. The economic cost of implementation is also considered using the Amazon cloud (EC2) pricing model.

Huang [46] proposed a low reflection ratio mitigation system to be deployed in front of the IaaS. The system consists of Source Checking, Counting, Attack Detection, Turing test and Question Generation modules. In the implementation of their defence system, the authors take into account the challenges of computational efficiency and overheads and their effect on legitimate users. The Turing test is embedded in the kernel and uses text-based questions generated using Lexical Function Grammar (LGF). This approach requires less bandwidth than the more traditional image-based puzzles, such as CAPTCHA. A blacklist, whitelist, block list and unknown are used to categorise incoming packets based on IP addresses. These lists are maintained by administrators through APIs. The use of these APIs opens the system to malicious manipulation from insiders. Although this system uses the cloud capabilities to provide protection against bottlenecks, it incurs an operational degradation of 8.5% when monitoring traffic against a blacklist of 100000 addresses.

Fujinoki et al. [47] build on the limitations of overlay networks in hiding the location of target servers through the use of gateway routers. The suggested Dynamic Binary User Splits (DBUS) system protects clouds from insider attacks and compromised user host machines. DBUS avoids the need for migration of network items to other hosts. It also removes the need to monitor all network traffic, which provides lower computational overhead. Each proxy router contains a Bloom filter, which is a data structure that can efficiently test for the presence of certain values. A user management table is used to hold records of which users are assigned to which proxy nodes and no user can return to a previous proxy once they have migrated. When an attack warning is issued, more user proxy

machines are deployed with the number of users assigned to each being halved until attackers are identified. Simulations results proves the DBUS a promising system. Yet, there is a need for this system to be implemented in a real-world environment and evaluated under real-world conditions.

Tripathi et al. [48] investigates the use of the open-source tool Hadoop in providing a DDoS defence. Hadoop provides tools that use the MapReduce framework for processing large amounts of data in association with the Hadoop Distributed File System (HDFS). The authors approach overrides the traditional First in First Out (FIFO) scheduling mechanism with a Self-Adaptive MapReduce (SAMR) scheduling algorithm, which divides jobs into tasks that are then assigned to map nodes. The benefit of SAMR is that it also reads the historical information that is stored on each node and adjusts its task distribution based on this information. By measuring nodes performance, the task execution time can be improved by up to 25%. Hadoop is capable of efficient network behaviour analysis. However, its current implementation needs further optimisation to be used for cloud defence.

An alternative method for detecting DDoS flooding attacks is presented in [49]. A distance estimation technique is used to estimate traffic rates. The distance value is calculated using the Time-To-Live (TTL) of a packet. The majority of operating systems only accept initial TTL values of 30, 32, 60, 64, 126 and 255 making the estimated distance the smallest of these values that is greater than the current TTL value. Exponential smoothing is then implemented to provide the real-time measurement of the roundtrip of IP traffic. Finally, absolute deviation is used to determine if the behaviour is abnormal. This approach attempts to avoid the reliance on attribute dependences that can be spoofed or the time delays associated with traffic monitoring. The authors suggest that ISPs should be responsible for implementing filters on traffic as they receive the packets. As previously discussed, this practice is unlikely to be adopted.

Latif et al. [50] review approaches to protect against DDoS attacks focussing on Wireless Body Area Networks (WBANs). WBANs devices are limited in computational power, available bandwidth, security and battery life. This makes them ideally partnered with the cloud, where much of the complex computational requirements can be moved. Due to resource restrictions, there is a need for minimal overheads in any WBAN DDoS defence system. A number of approaches has been proposed to address this requirement. For example, the system described in [51] places IDS' at different locations in the cloud space and then has them collaborate to share attack alerts. This approach assumes that a node will have the available bandwidth to send an alert when it is under attack. In [52], a simple IDPS that uses a statistical method to create and apply a covariance matrix of network behaviour is proposed. Current network behaviour is compared to the created model, while the TTL of packets is used to identify the source of the attack. In [53], a similar behaviour-based system featuring a training period is presented. The system computes a score for each packet. These scores are then used to determine which packets to drop in an attack scenario. This approach delivers a high-speed system with minimal memory requirements and an acceptable level of filtering accuracy. This makes it suitable for real-time implementation. A correlation pattern detection module was added to this system by Priyanka et al. [54] to overcome the flaws in the Confidence-based Filtering (CBF) by introducing a confidence value to the packet header.

As shown in this review of the recent IDPSs, there have been many suggestions for tackling DDoS attacks against the cloud computing paradigm. The cloud architecture pose many security vulnerabilities at different level, which resulted in solutions being proposed to primarily defence against a single type of or point of attack. An important step forward is the utilisation of the cloud capabilities in the design of defence systems. This enabled systems to adopt the scalability properties of the cloud to enhance the security for all parties. It is important for security solutions to provide protection for individual clients and their services as well as the cloud as a whole. Commonly, individual client applications and web services will be the targets of DDoS attacks. Individual clients will only notice issues with their own QoS and these are the issues that will further perpetuate the security fears of adopting cloud computing architectures. To develop a comprehensive defence system, aspects of these research solutions need to be integrated in one product to protect against a wider range of attacks. System designers should pay particular attention to the 'secure' integration of the cloud underlying technologies to avoid introducing further vulnerabilities to the cloud architecture.

VIII. Summary of Features – DDoS Cloud Protection Systems

In this section, we present a summary of all man concepts discussed in previous sections. The reviewed different attacks are listed with their corresponding response mechanisms. The recent solutions for various security issues are also grouped into logical categories to make gaps in the literature more obvious. This is followed by a taxonomy that attempts to classify DDoS protection systems, which have been proposed for the cloud computing paradigm. A description of each classification category and the order in which they have been applied is given in Table IV.

Figure 2, translates the taxonomy given in Table IV to a flow chart showing the exiting DDoS cloud protection systems and comparing the implementation of different features in the proposed systems. This allows us to see common techniques and highly secure facets of the systems, while also highlighting weaknesses and the areas of focus for future work in these areas.

TABLE III. SUMMARY OF CLOUD PROTECTION SYSTEMS KEY FEATURES

A. **Intrusion Detection** Description: All systems identified in this survey make use of intrusion detection to raise an alarm and to determine which response to adopt. Currently, statistical models are the most widely used methods for detecting intrusions. Other methods include the use of Artificial Intelligence and Neural Networks.	
Knowledge-based IDS	• Used for identifying previously known attacks. • Attack signatures stored in databases. • Simple to implement. • Require frequent updates to maintain security level. • Slow to react to new attacks, as new behaviours signatures need to be added to the relevant databases. • The aim is to define suitably abstract signatures that can potentially recognise new attack designs based on previously identified patterns. • Provides effective attack classification, which allows a more targeted response to be triggered.
Anomaly-based IDS	• Monitors network attributes. • Assumes that malicious network behaviour is noticeably different to regular behaviour. • Able to detect unknown attacks. • The usability of these systems is dependent on the false alarm rate. • Requires a system-training period. The training period needs to be carefully selected to represent standard network behaviours. • Models can be updated with new information while deployed. This may be required as business practices evolve. • Data analysis tools, such as Hadoop, can be used to create models and monitor real-time behaviour [48]. • Suitable behaviour models are difficult to create given the user flexibility that the cloud introduces. • Can struggle to classify the type of attack meaning that only generalised responses can be issued. • Greater implementation complexity.
Hybrid	• A combination of knowledge-based and anomaly-based IDSs used to combine the strengths of both of types of approach. • Highest level of complexity to implement. • Higher overheads as packets are monitored through multiple types of IDS.
Deployment	• Deploying IDSs close the server makes identification of attacks easier. It also limits the effectiveness of countermeasures. • Deploying IDS further from the server or external to a perimeter firewall makes the identification of an attack more complex. However, this allows countermeasures to have a greater impact.
B. **Responses** Description: The reaction of a system once an intrusion or attack has been detected. This can involve responses to neutralise an individual attack and the introduction of preventative measures to secure the system against future attacks of the same nature. These responses have been summarised into categories.	
Filtering	• Update upstream filters to block traffic once the source is identified. This assumes that there is enough bandwidth to send these messages. This may be compromised in a DDoS attack scenario. • Deployment locations can greatly affect response outcome. • Update firewall protocols to include new responses. • Add information to packet headers to identify legitimate packets [19].
Rate Limiting	• Attempts to relieve the pressures on bottlenecks. • This affects all network traffic, not just the malicious.
Adapt use of Virtual Machines	• Using cloud features (scalability) to increase/decrease number of VMs as required [17]. • Increase the number of VMs to enhance isolation (DBUS - [47]). • Logical and physical migration of resources [8, 45].
Identify attack source	• Use trace-back techniques to identify the source of attacks [39]. • Trace-back systems have limited effectiveness against multisource distributed attacks • Can fail to identify 'spoofed' IP addresses.

C. **Management** Description: How are different aspects of the proposed system managed to ensure security.	
Authentication	• Strong authentication for legitimate users. Using image-based (CAPTCHA) or text-based puzzles [46]. • Authentication needs to avoid being obstructive to the user experience. For example, puzzles are suitable for logins but should not be used for routine tasks.
System Monitoring	• Several systems use VMMs to monitor the state of deployed VMs. Having these dedicated machines is a useful consideration but they risk becoming a bottleneck in high traffic situations [34]. • Using the scalability features of the cloud to deploy further VMMs as required can reduce the risk of bottlenecks [8].
Overheads	• The majority of systems aim to scan each packet as it enters the cloud network. This can introduce large overheads into the system affecting the users QoS. If only a single machine is allocated the task of packet monitoring, this can create a bottleneck in the system. • To reduce detection overheads, many systems aim to detect a limited set of attack traits.

TABLE IV. DDOS CLOUD PROTECTION SYSTEM TAXONOMY

<table>
<tr><th colspan="2">Taxonomy Layers</th></tr>
<tr><td>IDPS</td><td>There are two fundamental approaches to intrusion detection used by the protection system. These are typically either knowledge-based that use signatures to recognise attacks that have previously occurred, or anomaly-based that make use of data models to identify suspicious network behaviour. Anomaly-based systems have the advantage of being able to identify previously unseen attacks; however, they can suffer with high false positive rates.
<table>
<tr><th>Knowledge-based Intrusion Detection</th><th>Anomaly-based Intrusion Detection</th></tr>
<tr><td>• [19] Karnwal (2012)
• [46] Huang (2013)
• [47] Fujinoki (2013)
• [17] Yu (2014)
• [48] Tripathi (2013)
• [54] Priyanka (2013)</td><td>• [39] Joshi (2012)
• [40] Vissers (2014)
• [41] Shtern (2014)
• [8] Wang (2014)
• [45] Jia (2014)
• [49] Chopade (2013)
• [52] Ismail (2012)
• [53] Chen (2011)</td></tr>
</table></td></tr>
<tr><td>VM Management Point</td><td>Compares the use of VMM modules to manage systems. Classification is based on whether a single VMM is used to monitor VMs regardless of the scale of the cloud resources being used, or whether the number of VMMs is increased when required based on cloud scaling principles.
<table>
<tr><th>Scalable VMM system used (Distributed)</th><th>Single VMM used (Centralised)</th></tr>
<tr><td>• [17] Yu (2014)
• [41] Shtern (2014)
• [8] Wang (2014)
• [45] Jia (2014)</td><td>• [39] Joshi (2012)
• [40] Vissers (2014)
• [49] Chopade (2013)
• [52] Ismail (2012)
• [53] Chen (2011)
• [19] Karnwal (2012)
• [46] Huang (2013)
• [47] Fujinoki (2013)
• [48] Tripathi (2013)
• [54] Priyanka (2013)</td></tr>
</table></td></tr>
</table>

User Authentication

What form of user authentication is employed by the system? Many systems incorporate authentication protocols to identify legitimate users. Typically, strong user authentication involves puzzles such as CAPTCHA. Other methods include marking packets, but these can suffer from spoofing attacks. Below is a classification of recent solutions by whether or not the system uses puzzles for user authentication.

Yes	No
• [47] Fujinoki (2013) • [48] Tripathi (2013) • [54] Priyanka (2013) • [40] Vissers (2014)	• [39] Joshi (2012) • [49] Chopade (2013) • [52] Ismail (2012) • [53] Chen (2011) • [19] Karnwal (2012) • [46] Huang (2013) • [17] Yu (2014) • [41] Shtern (2014) • [8] Wang (2014) • [45] Jia (2014)

Response

What is the response method of the proposed system? The range of response methods is quite varied and can consist of several layers but can be grouped into categories. Responses include trace back techniques to discover the source of techniques, which can then be used to update filters. Other responses include techniques used to filter and drop packets and user requests, and those that migrate users to new VMs when current ones become compromised.

Trace back	VM Migration	Filters/Block lists/Dropped Packets
• [19] Karnw (2012) • [39] Joshi (2012)	• [41] Shtern (2014) • [8] Wang (2014) • [45] Jia (2014) • [47] Fujinoki (2013)	• [46] Huang (2013) • [49] Chopade (2013) • [40] Vissers (2014) • [53] Chen (2011) • [52] Ismail (2012)

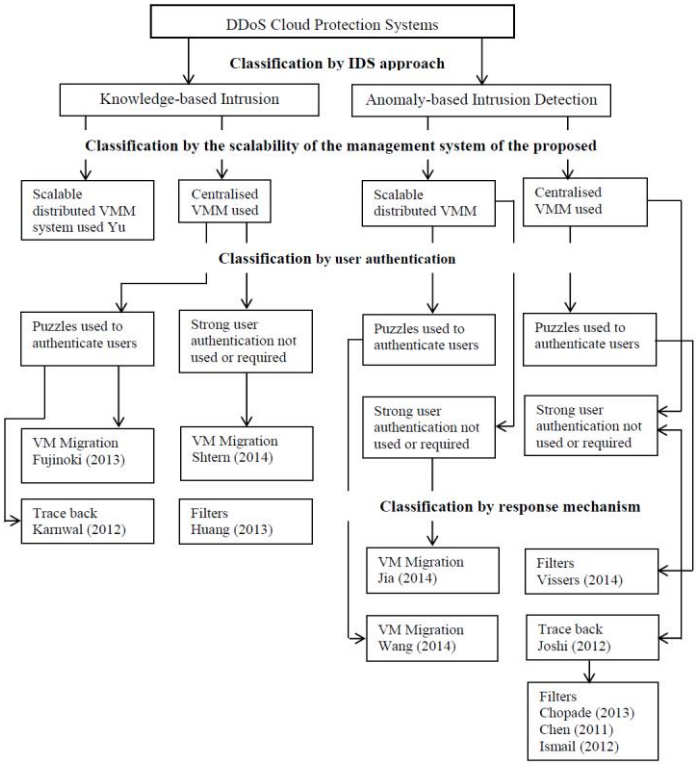

Fig. 2. DDoS cloud protection systems

IX. CONCLUSION

This paper examined the latest security issues in virtualisation technologies and IDPS to defend against DDoS attacks targeting cloud systems. Based on this comprehensive review, it is apparent that approaches to security across all reviewed areas share many common themes. These themes include:

1) Who and how will the security system be managed and monitored?

2) How will alerts be triggered?

3) How is the false alarm rate reduced?

4) What is the impact on operational overheads?

With virtualisation being a key underlying technology of the cloud computing paradigm it is understandable that there are a number of similarities between proposed systems. A number of these highlight the use of VMs as system management units and a few of these allow these to be generated in a similar way to user VMs in the cloud. This allows these systems to make use of the elasticity and scalability of the cloud paradigm to provide a more effective response to an attack and helps to reduce bottlenecks in the system.

A common response to an attack is to migrate users to new VMs through either physical or logical migration. This uses the strengths of VMs and helps to enforce the principles of isolation. Systems must be in place to remove compromised VMs to ensure that they are not migrated along with other users.

Commonly, the response of the system is designed based on non-attack or low-level attack conditions. This allows systems, even under test conditions to deliver the necessary messages to update filter protocols and perform other defensive manoeuvres. It must be considered, that under high stresses these systems may not operate in the same manner. Against a DDoS flooding attack, it may not be possible to update upriver

filters effectively enough to reduce the intensity of the attack. Proposed systems must therefore test themselves under such strains so that their behaviours at these attack intensities can be observed.

The majority of defence systems focus on a particular type or point of attack against which they can be shown to be effective. The next step is to integrate these approaches to provide a more universal protection. When implementing this integration it is important that new vulnerabilities are not introduced into the system.

In the authors opinion, there are two research main research avenues to be followed. In the first, the intrusion is attempting to compromise VMs in order to launch a DDoS attack against a target that is external to the cloud. Once the intrusion is detected, a counter-measure is to be deployed, which in this case will be a calibrated firewall. Although this is may appear to be a simplistic fix to exiting protocols, it is not a solution that is widely adopted by current CSPs because it adds to their overheads, while not directly protecting their own infrastructure. The second considers a more traditional cloud intrusion where the target of the attack is the cloud or an element within the cloud itself. Resources relevant to this scenario will be based in a Eucalyptus cloud system.

REFERENCES

[1] M. Mackay, et al., "Security-oriented cloud computing platform for critical infrastructures," Computer Law & Security Review, vol. 28, pp. 679-686, 2012.

[2] C. Rong, et al., "Beyond lightning: A survey on security challenges in cloud computing," Comput. Electr. Eng., vol. 39, pp. 47-54, 2013.

[3] J. Li, et al., "CyberGuarder: A virtualization security assurance architecture for green cloud computing," Future Generation Computer Systems, vol. 28, pp. 379-390, 2012.

[4] J. Jang-Jaccard and S. Nepal, "A survey of emerging threats in cybersecurity," Journal of Computer and System Sciences, vol. 80, pp. 973-993, 2014.

[5] Galante J., et al. (2011, May 14, 2015). Sony network Breach shows Amazon Cloud's appeal for hackers. Available: http://bloomberg.com/news/2011-05-15/sony-attack-shows-amazon-s-cloud-service-lures-hackers-at-pennies-an-hour.html

[6] W. Wei. (2014, May 14, 2015). Sony Playstation Network Taken Down By DDoS Attack. The Hackers News. Available: http://thehackernews.com/2014/08/sony-playstation-network-taken-down-by_24.html

[7] M. Kumar. (2011, May 14, 2015). Cloud Computing Used to Hack Wireless Password. The Hackers News. Available: http://thehackernews.com/2011/01/cloud-computing-used-to-hack-wireless.html

[8] H. Wang, et al., "A moving target DDoS defense mechanism," Computer Communications, vol. 46, pp. 10-21, 2014.

[9] M. Darwish, et al., "Cloud-based DDoS attacks and defenses," in Information Society (i-Society), 2013 International Conference on, 2013, pp. 67-71.

[10] P. paganini. (2013, May 14, 2015). Cybercriminals using hijacked Cloud hosting accounts for targeted attacks. The Hackers News. Available: http://thehackernews.com/2013/06/cybercriminals-using-hijacked-cloud.html

[11] S. Khandelwal, "SNMP Reflection DDoS Attacks on the Rise. The Hackers News," 2014.

[12] A. Patel, et al., "An intrusion detection and prevention system in cloud computing: A systematic review," Journal of Network and Computer Applications, vol. 36, pp. 25-41, 2013.

[13] A. Rehman, et al., "Virtual machine security challenges: case studies," International Journal of Machine Learning and Cybernetics, vol. 5, pp. 729-742, 2014/10/01 2014.

[14] J. E. Smith and R. Nair, "The architecture of virtual machines," Computer, vol. 38, pp. 32-38, 2005.

[15] S. Subashini and V. Kavitha, "Review: A survey on security issues in service delivery models of cloud computing," J. Netw. Comput. Appl., vol. 34, pp. 1-11, 2011.

[16] B. Loganayagi and S. Sujatha, "Enhanced Cloud Security by Combining Virtualization and Policy Monitoring Techniques," Procedia Engineering, vol. 30, pp. 654-661, 2012.

[17] S. Yu, et al., "A Security-Awareness Virtual Machine Management Scheme Based on Chinese Wall Policy in Cloud Computing," The Scientific World Journal, vol. 2014, p. 12, 2014.

[18] Briscoe. (2011 Marinos: Digital Ecosystems in the Clouds: Towards Community Cloud Computing. Available: http://blog.ascens-ist.eu/wp-content/uploads/2011/03/xaas.png

[19] T. Karnwal, et al., "A comber approach to protect cloud computing against XML DDoS and HTTP DDoS attack," in Electrical, Electronics and Computer Science (SCEECS), 2012 IEEE Students' Conference on, 2012, pp. 1-5.

[20] R. Koch, et al., "Behavior-based intrusion detection in encrypted environments," Communications Magazine, IEEE, vol. 52, pp. 124-131, 2014.

[21] J. McHugh, "Intrusion and intrusion detection," Digital Object Identifier (DOI) 10.1007/s102070100001, pp. 14-35, : 27 July 2001 2001.

[22] S. M. Alqahtani, et al., "An Intelligent Intrusion Prevention System for Cloud Computing (SIPSCC)," in Computational Science and Computational Intelligence (CSCI), 2014 International Conference on, 2014, pp. 152-158.

[23] C. Modi, et al., "A survey of intrusion detection techniques in Cloud," Journal of Network and Computer Applications, vol. 36, pp. 42-57, 2013.

[24] S. X. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," Applied Soft Computing, vol. 10, pp. 1-35, 2010.

[25] D. Moore, et al., "Inferring Internet denial-of-service activity," ACM Trans. Comput. Syst., vol. 24, pp. 115-139, 2006.

[26] R. Goel, et al., "Cloud Computing Vulnerability: DDoS as Its Main Security Threat, and Analysis of IDS as a Solution Model," in Information Technology: New Generations (ITNG), 2014 11th International Conference on, 2014, pp. 307-312.

[27] H. Beitollahi and G. Deconinck, "Analyzing well-known countermeasures against distributed denial of service attacks," Computer Communications, vol. 35, pp. 1312-1332, 2012.

[28] A. Volokyta, et al., "Secure virtualization in cloud computing," in Modern Problems of Radio Engineering Telecommunications and Computer Science (TCSET), 2012 International Conference on, 2012, pp. 395-395.

[29] L. Qian, et al., "An In-VM Measuring Framework for Increasing Virtual Machine Security in Clouds," Security & Privacy, IEEE, vol. 8, pp. 56-62, 2010.

[30] D. Williams, et al., "Security through Diversity: Leveraging Virtual Machine Technology," Security & Privacy, IEEE, vol. 7, pp. 26-33, 2009.

[31] O. Al-Jarrah and A. Arafat, "Network Intrusion Detection System using attack behavior classification," in Information and Communication Systems (ICICS), 2014 5th International Conference on, 2014, pp. 1-6.

[32] L. Laboratory. (2014, DARPA INTRUSION DETECTION EVALUATION. Available: http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/index.html

[33] C. Chun-Jen, et al., "NICE: Network Intrusion Detection and Countermeasure Selection in Virtual Network Systems," Dependable and Secure Computing, IEEE Transactions on, vol. 10, pp. 198-211, 2013.

[34] H. Jin, et al., "A VMM-based intrusion prevention system in cloud computing environment," The Journal of Supercomputing, vol. 66, pp. 1133-1151, 2013/12/01 2013.

[35] A. Alhomoud, et al., "Performance Evaluation Study of Intrusion Detection Systems," Procedia Computer Science, vol. 5, pp. 173-180, 2011.

[36] H. Mohamed, et al., "A collaborative intrusion detection and Prevention System in Cloud Computing," in AFRICON, 2013, 2013, pp. 1-5.

[37] A. Alazab, et al., "Using response action with intelligent intrusion detection and prevention system against web application malware," Information Management & Computer Security, vol. 22, pp. 431-449, 2014.

[38] Y. Lanjuan, et al., "Defense of DDoS attack for cloud computing," in Computer Science and Automation Engineering (CSAE), 2012 IEEE International Conference on, 2012, pp. 626-629.

[39] B. Joshi, et al., "Securing cloud computing environment against DDoS attacks," in Computer Communication and Informatics (ICCCI), 2012 International Conference on, 2012, pp. 1-5.

[40] T. Vissers, et al., "DDoS defense system for web services in a cloud environment," Future Generation Computer Systems, vol. 37, pp. 37-45, 2014.

[41] M. Shtern, et al., "Towards Mitigation of Low and Slow Application DDoS Attacks," presented at the Proceedings of the 2014 IEEE International Conference on Cloud Engineering, 2014.

[42] P. C. Senthilmahesh, et al., "DDoS Attacks Defense System Using Information Metrics," in Proceedings of the Third International Conference on Trends in Information, Telecommunication and Computing. vol. 150, V. V. Das, Ed., ed: Springer New York, 2013, pp. 25-30.

[43] R. Mathew and V. Katkar, "Survey of low rate DoS attack detection mechanisms," presented at the Proceedings of the International Conference & Workshop on Emerging Trends in Technology, Mumbai, Maharashtra, India, 2011.

[44] Y. Tang, "Countermeasures on Application Level Low-Rate Denial-of-Service Attack," in Information and Communications Security. vol. 7618, T. Chim and T. Yuen, Eds., ed: Springer Berlin Heidelberg, 2012, pp. 70-80.

[45] J. Quan, et al., "Catch Me If You Can: A Cloud-Enabled DDoS Defense," in Dependable Systems and Networks (DSN), 2014 44th Annual IEEE/IFIP International Conference on, 2014, pp. 264-275.

[46] V. S. Huang, et al., "A DDoS Mitigation System with Multi-stage Detection and Text-Based Turing Testing in Cloud Computing," in Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on, 2013, pp. 655-662.

[47] H. Fujinoki, "Dynamic Binary User-Splits to Protect Cloud Servers from DDoS Attacks," presented at the Proceedings of the Second International Conference on Innovative Computing and Cloud Computing, Wuhan, China, 2013.

[48] S. Tripathi, et al., "Hadoop Based Defense Solution to Handle Distributed Denial of Service (DDoS) Attacks," Journal of Information Security, 2013.

[49] S. S. Chapade, et al., "Securing Cloud Servers Against Flooding Based DDOS Attacks," in Communication Systems and Network Technologies (CSNT), 2013 International Conference on, 2013, pp. 524-528.

[50] R. Latif, et al., "Distributed Denial of Service (DDoS) Attack in Cloud-Assisted Wireless Body Area Networks: A Systematic Literature Review," J. Med. Syst., vol. 38, pp. 1-10, 2014.

[51] L. Chi-Chun, et al., "A Cooperative Intrusion Detection System Framework for Cloud Computing Networks," in Parallel Processing Workshops (ICPPW), 2010 39th International Conference on, 2010, pp. 280-284.

[52] M. N. Ismail, et al., "Detecting flooding based DoS attack in cloud computing environment using covariance matrix approach," presented at the Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication, Kota Kinabalu, Malaysia, 2013.

[53] C. Qi, et al., "CBF: A Packet Filtering Method for DDoS Attack Defense in Cloud Environment," in Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on, 2011, pp. 427-434.

[54] N. Priyanka, et al., "Enhanced CBF Method to Detect DDoS Attack in Cloud Computing Environment," International Journal of Computer Science Issues, IJCSI, vol. 10, pp. 142-146, 2013.