System and Network Engineering

IoT (D)DoS prevention and corporate responsibility: A model to prevent polluting the internet

University of Amsterdam
Swann Scholtes
swann.scholtes@os3.nl

Supervisor (KPMG)
Alex Stavroulakis
Stavroulakis.Alex@kpmg.nl

Supervisor (KPMG)
Can Kurnaz
Kurnaz.Can@kpmg.nl

Assessor (UvA)
Prof. dr. ir. Cees T.A.M. de Laat
rp@os3.nl

June 21, 2019

Faculty of Science
University of Amsterdam

Abstract – Over the past decade IoT is slowly becoming more inter weaved with traditional IT networks. This development of IoT also brings with it the threats and dangers that traditional networks have. In particular denial of service (DoS) attacks are of concern. Not only are IoT devices targeted by DoS attacks, they are also a source to launch DoS attacks from. Moreover, IoT devices often have improperly configured devices with respect to security, which leads to more and more IoT devices being used in DoS attacks. This in combination with the rapid growth of Internet of Things (IoT) is a concerning development. As a result governments are exploring legislative options to combat this phenomenon. In this paper we present a model for IoT architectures which can be utilised to prevent (D)DoS traffic originating internally from reaching the internet. The model uses an IoT architecture separated by an object, network, support and application layer. Our model specifies an objects, intermediate and edge defensive layer where mitigation strategies are executed. Thereby, preventing possible liability claims.

Contents

1 Introduction

Internet of things (IoT) is used in a plethora of environments like, home user systems, business IT, healthcare, and many more. Over the past decade IoT is slowly becoming more inter weaved with traditional IT networks. This development of IoT also brings with it the threats and dangers that traditional networks have. In particular denial of service (DoS) attacks are of concern. Not only are IoT devices targeted by DoS attacks, they are also a source to launch DoS attacks from. Moreover, IoT devices often have improperly configured devices with respect to security [3] [10], which leads to more and more IoT devices being used in DoS attacks. This is a dangerous development as can be seen from the Dyn DoS attack of late 2016 where critical infrastructure was rendered unavailable through the use of a DoS attack. Therefore the need arises to better detect and prevent DoS attacks launched from IoT environments as well as minimize the potential contribution of said IoT environments.

1.1 Motivation

In this subchapter we talk about recent developments in the field of IoT and DoS. In particular we highlight major DDoS attacks, IoT manufacturers, IoT devices deployed and the growth aspect pertaining to the IoT landscape.

1.1.1 Major DDoS attacks

The European Union Agency for Network and Information Security (ENISA) published an article where they highlight major DDoS attacks involving IoT devices [9]. On 20 September 2016 there was a 620Gbps DDoS attack on KrebsOnSecurity.com. The website was protected pro bono by Akamai, though Akamai dropped the DDoS protection due to the scale of the attack. Furthermore, Akamai noted that the attack was launched from compromised IoT devices. Just a few days later on 22 September 2016 the hosting provider OVH was also hit by a DDoS attack. This attack was 990Gbps and was mainly launched from 145,000 compromised IoT IP cameras and DVR players. On 21 October 2016 the DNS provider Dyn was attacked by a DDoS attack which had far reaching consequences. The attack itself reached speeds of 1.2Tbps. For several hours customers from Dyn where unable to reach popular websites like Twitter, Amazon, Tumblr, Reddit, Spotify and Netflix. Dyn stated that the attack was carried out by approximately 100,000 compromised devices. The largest DDoS attack registered was the 2018 February DDoS against Github [7]. This attack reached speeds of up to 1.3Tbps. Furthermore, an alleged 1.7Tbps DDoS attack was lauched 5 days after the attack on Github, though the victim was never publicly disclosed making it hard to verify.

1.1.2 IoT devices deployed

The IoT devices deployed internal and in the wild is growing at a rapid rate. Gartner identifies growing trends of deployed IoT devices in their press release of late 2018 [12]. Gartner states that by the end of 2019, 14.2 billion "things" will be in use. Furthermore, by 2021 this number will reach 25 billion, which is an increase of 76.05% in just 2 years.

1.1.3 IoT manufacturers

IoT manufacturers often develop their equipment and devices to be first to market [10]. The consequence is that the security of the device is usually an afterthought and is not designed with security in mind. In other words the IoT manufacturers often care about availability and not so much about confidentiality or integrity. Due to this negligence multiple different proposals have been made, including legislative proposals. Viktor Vitkowsky stated that the Federal Trade Commission (FTC) could hold IoT manufacturers liable for violating section 5 of the FTC [34]. Section 5 states that IoT devices with security vulnerabilities could be used in DoS attacks. IoT manufacturers can be prosecuted based on section 5 of the FTC for inadequately securing their devices. This could cause other IoT manufacturers to recall their devices to adequately secure them. Though this only has an effect where the FTC has jurisdiction. Viktor Vitkowsky also noted that businesses damaged by IoT launched DDoS attacks could bring civil claims. In these cases product liability will be enforced in U.S. courts. Senator Mark R. Warner asked the Federal Communications Commission (FCC) if Internet Service Providers (ISPs) could be required to mark vulnerable IoT devices as insecure and prevent them from accessing the internet [35]. Furthermore, Senator R. Warner asked the FCC if IoT manufacturers should abide by minimum technical security standards defined by the FCC [35].

Likewise, the "Nederlandse Norm" (NEN) states that there is a lack of standardisation which is further strengthened by the "Agentshap Telecom" though they assume a passive position and let the market come up with standardisation standards themselves [20]. Furthermore, after the 2016 Dyn DDoS attack Kees Verhoeven of the house of representatives asked question about a possible quality mark or control stamp to verify if IoT devices have proper security implementations. Moreover, in June 2018 there were questions of the house of representatives [14]. The house of representatives asked the Ministry of Justice and Security when a quality mark or control stamp is ready to be deployed and how the process is proceeding. Furthermore, the house of representatives asked the Ministry of Justice and Security if internet service providers (ISP) and telecommunication companies have enough capabilities to detect insecure IoT devices, implying the possibility of sanctions against said companies.

1.1.4 Attack growth

An attacker has to compromise an IoT device before the attacker can execute malicious actions from the device. Malware is one of the possible routes an attacker could take. Mikhail Kuzin et al. together with Kaspersky Lab created an overview of IoT malware samples encountered in the years 2016, 2017 and 2018 [22]. 3,219 malware samples where encountered in 2016 by Kaspersky Lab, while 32,614 malware samples where registered in 2017. This is an increase of 1,013.17%. In 2018 Kaspersky Lab encountered 121,588 malware IoT samples. This is an increase of 372.80% compared to 2017. Alibaba regards this increase to the fact that Mirai botnet source code was published late 2016 [6] and is being used in new derivatives. One way to gain access to IoT devices is by cracking the login credentials of the device. Kaspersky Lab [22] states that the login cracking attacks are mostly carried out against the Telnet and SSH protocol, which are 75.40% and 11.59% respectively.

The three leading countries of infected devices are Brazil, China and Japan with respectively 23%, 17% and 9%. Gurubaran et al. created an overview of the top DDoS attack vectors as depicted in Figure 1 [30]. The overviews shows that the most used attack vectors are the UDP, TCP SYN and ICMP attacks. The UDP attack is by far the largest with a share of 31.56% followed by the TCP SYN attack with 18.50%. ICMP attacks are responsible for 9.32% of the total attacks. Furthermore, attacks larger than 300Gbps are always pure TCP SYN attacks or a blend of different attack vectors including TCP SYN. Oleg Kupreev et al. comprised a chart which shows the top countries where DDoS attacks are launched from [26]. The top three countries are China, United States and Australia. China scored 43.26% down from 70.58% while the United States scored 29.14% up from 17.05%. Australia scored 4.57% up from 2.27%. Furthermore, the ratio of Windows and Linux botnets almost did not change. Botnets are still mainly active on the Linux platform which has a share of 97.11% up from 95.86%. The top 3 countries where command and control server (C&C) are located are the United States, Great Britain and the Netherlands. The Unites States has a share of 43.48% up from 37.31% while Great Britain has a share of 7.88% up from 3.73%. The Netherlands saw a growth of C&C server hosting, taking a share of 6.79% up from 2.24%. This is an increase of 303.12% from 2018 Q3 to 2018 Q4.

2 Research Questions

To achieve our goal, we deduced the following research question:

- How can organisations prevent contributing to Internet of Things denial of service attacks?

Based on that question, the following sub-questions were deduced:

1. What technical denial of service detection methods can organisations employ in Internet of Things networks?

2. What technical denial of service prevention methods can organisations employ in Internet of Things networks?

3. What measurements in Internet of Things networks minimise contributing to denial of service attacks?

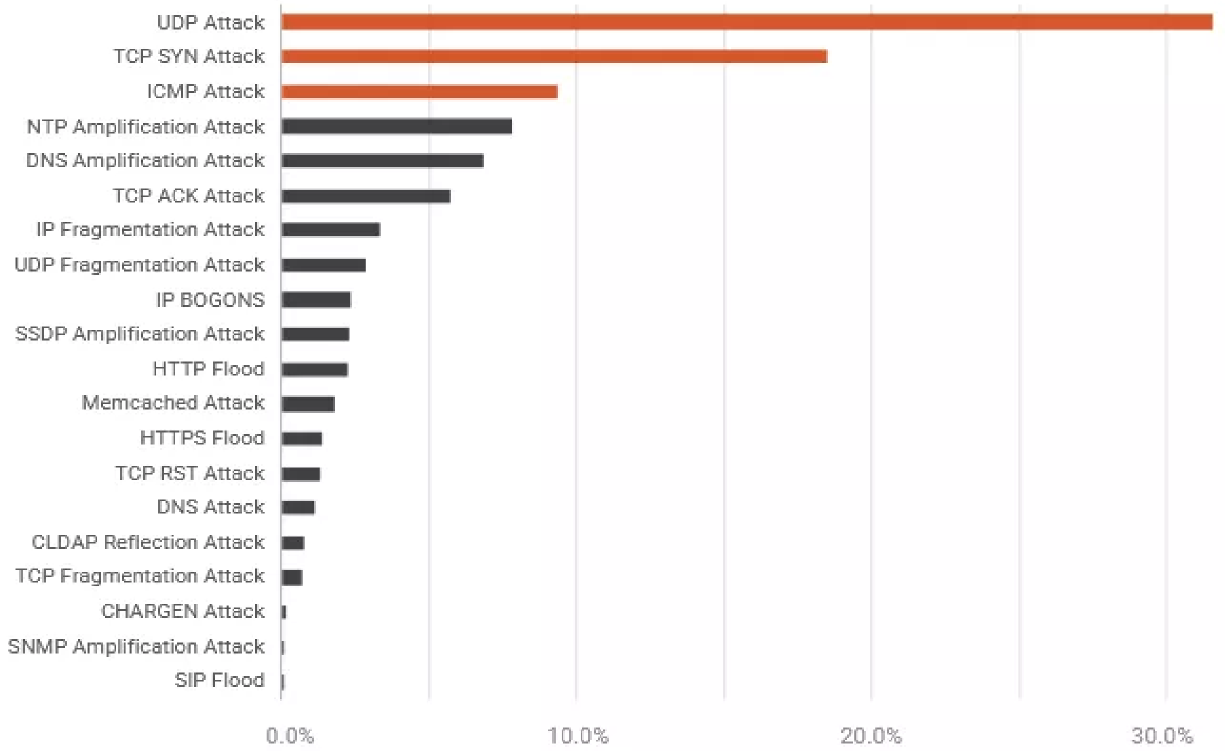

Figure 1: DDoS attack types [30].

3 Background

In this section we familiarise the reader with the topics at hand. This section is further divided into two main topics involved, namely IoT and DoS. Both topics have a subsection dedicated to them.

3.1 Internet of Things

In this subsection we highlight the IoT architecture including gateways, IoT connections, protocols and IoT devices.

3.1.1 IoT devices

Keshav Sinha et al. [32] categorized IoT devices into two different categories, namely full function device (FFD) and reduced function device (RFD). FFD devices are usually connected to main power and have enough resources to run traditional IT solutions like encryption algorithms. A router can be classified as a FFD device. RFD devices have reduced functionality and usually can not run traditional IT solutions due to insufficient power, processing power and memory constraints. RFD devices can also work with batteries or use rechargeable energy.

3.1.2 IoT connections and protocols

IoT devices can use a multitude of different protocols to communicate with each other. In this subsection we discuss the most prevalent protocol families used in IoT today. Wired connections are widely used in traditional IT. The Institute of Electrical and Electronics Engineers (IEEE) specifies the Ethernet protocol in the IEEE 802.3 specification [19]. IoT devices can use Ethernet as their communication protocol, though they need a full TCP/IP stack to properly communicate with other devices. Therefore, wired Ethernet connection are usually only implemented on FFD devices. Wireless is another communication option. IEEE published a wireless standard defined in the 802.11 family [17]. Traditional wireless protocols are resource intensive on the power side due to the antenna needing power to send the data into the used spectrum. This power requirement invalidates most RFD devices since they are scares on power and often use batteries or rechargeable energy. However, for FFD devices this is a suitable solution.

A standard for low-rate wireless personal area networks is defined by IEEE in 802.15.4 [18]. The 802.15.4 standard redefines the physical and medium access control layers of the Open Systems Interconnection model (OSI). With these adjustments low power devices can still use wireless technologies. The layers above physical and medium access control are from another protocol. Multiple different protocols use the 802.15.4 standard as the basis for theirs. We will not go into detail of these protocols but they include Zigbee, ISA100.11a, WirelessHART and many more.

3.1.3 IoT architecture

Multiple papers define an IoT architecture divided by layers [5][10][1]. In general they specify 3 main layers. The bottom layer is called the perception layer or objects layer. In this layer the sensors and actuators are found. The sensors and actuators are connected to a controller or coordinator. The controller is a device which acts as a central hub where traffic from the relevant sensors and actuators flows through. We differentiate mainly between IP networks and non IP networks. The controller is responsible to convert non IP traffic to IP traffic. However, there exists a multitude of different protocols that IoT devices can use. This is usually dependant on the manufacturer of the IoT devices. The middle layer is called the network layer or transport layer. This layer is responsible for transporting the data between the objects layer and the higher application layer. The traffic from the objects layer goes through a gateway in the network layer. The gateway is a central point where all traffic from the objects layer flows through. Therefore, access management is applied at the gateway. The network itself can be connected to the internet or internal only. The top layer is the application layer. The application layer can be depicted with a sub layer support or platform. In the support layer the data send from the objects layer through the network layer is pre-processed before it reaches the applications in the application layer. As a final step the relevant data is shown on the application. A support layer is usually only needed if the data is too much to handle for the individual application. The support layer can include Cloud, Intelligent or For computing.

3.2 Denial of Service

In this sub section we give the reader background information about DoS. More specifically we talk about infection, attack infrastructures, attack types, detection and mitigation.

3.2.1 Infection

An IoT device has to be infected before malicious actions can be performed. Fatima Hussain et al. defined two general ways devices get infected [16]. Devices get infected through an active approach or a passive approach. With the active approach an attacker is actively scanning the network for vulnerable devices. The payload is sent after the attacker has identified the vulnerable devices, which in turn gives the attacker access to the device. At this point the attacker can execute malicious action on the network. An attacker uses the passive way to let devices or users infect themselves. There is a multitude of different ways to passively infect devices. Attackers can forge e-mail messages and include malware in well-known document types which are likely to be opened by the receiver. The attack can also take control of an update server where he replaces the firmware of devices with malicious content. Devices get served the malicious software instead of the proper update if devices use the compromised server. After updating from the compromised server the attacker gains access to the device in question and again can execute malicious actions on the network.

3.2.2 Attack infrastructures

Attackers need an infrastructure to manage their infected devices. These infected devices are often called "zombies" or "slaves". Rashid Ali Mirza et al. describes an attack infrastructure that attackers use to launch DDoS attacks from [23]. Attackers manage their infected devices through a command and control server (C&C). A C&C server can send instructions to the infected devices, triggering them to perform malicious actions like launching a DDoS attack. The C&C server is often a central point in the attack infrastructure. Without a C&C server attackers can not control their infected devices anymore.

3.2.3 DoS attacks

There exists multiple different DDoS attacks. Ahmad Riza'ain Yusof et al. categorizes these attacks into categories [37]. They are volumetric, amplification, protocol and application attacks. An attacker uses the available bandwidth from their botnet to overwhelm the victim with traffic. The victim goes offline if the attacker has more bandwidth available than the victim. An attacker can increase his bandwidth by utilizing amplification. An attacker can send his traffic to a server which amplifies the request and sends the response to the victim. A key property of amplification servers is that the response is bigger than the question. Therefore, the available bandwidth of the attacker increases making it easier to overwhelm a victim.

Attackers can also exploit the protocol that is used at the victim side. An example is the TCP SYN attack where attackers create half open connections draining resources of the victim server. The service offered by the victim server goes offline if the resources are depleted creating a denial of service. Another example is the ping of death attack where the packet length exceeds the maximum length specified in the standard. At the victim side this packet is interpreted wrongly and causes the system to crash. Application attacks are focused on the end applications themselves. These attacks are the hardest to identify as stated by Ahmad Riza'ain Yusof et al. [37]. In general, application attacks try to depleted the resources on the victim server. An example of this is session flooding, creating a multitude of sessions until the server resources are depleted causing a denial of service.

3.2.4 Detection

Detection mechanisms of traditional Intrusion Detection Systems (IDS) can be divided into two categories as demonstrated by Omar E. Elejla et al. [8]. One approach is signature based detection. Signatures are created from known attacks similar to how antivirus programs work. The detection rate of signature based detection is generally high. However, attacks where no signature has been created beforehand are not detected. Another method is anomaly based detection. An IDS needs to learn the normal working state of a network. An IDS using anomaly detection then flags everything that does not correspond to the mapped network. An advantage with this approach is that new attacks can also be detected. A negative effect of this approach is the false positives that anomaly detection can generate. Another method to detect DoS attacks is threshold based approaches as described by Kanwalvir Singh et al. [31]. Threshold based detection uses the available metrics of network equipment to make decisions about DDoS attacks. Metrics can include but are not limited to packet length, packets per second and bandwidth usage.

3.2.5 Mitigation

There are different mitigation techniques available. A well known solution including access control lists (ACL) and blackholing is described by Triantopoulou Stamatia et al. in [33]. The ACL blackholing approach can be used in a manual fashion or automatic with the use of scripting. A negative effect of blackholing is that legitimate traffic will also be blocked alongside the malicious traffic. Commercial mitigation solutions are also available in the form of scrubbing appliances. Eric Osterweil et al. shows that scrubbing appliances are commercial devices that are placed at strategic places in the network [27]. The malicious traffic flow is redirected towards the scrubbing appliance after a DDoS attack is detected. The scrubbing appliance then cleans the traffic of any malicious content and returns the cleaned traffic towards the network. Third party offsite solutions are available in the form of scrubbing centres. An example is the NaWas solution offered by the "nationale beheerorganisatie internet providers" (NBIP) [24]. An organisation sends the malicious traffic to the scrubbing centre after they detect that a DDoS attack is ongoing. The scrubbing centre removes any malicious traffic and redirects the cleaned traffic back to the organisation. A peering relationship between the scrubbing centre and organisation is preferred due to the volume of traffic that DDoS attacks can generate.

3.3 Related work

Numerous research papers describe different security attacks in IoT environments. Muhammad Umar Farooq et al. and Antoine Gallais et al. list different IoT security attacks including DoS attacks [10] [11]. Furthermore, Mukrimah Nawir et al. shows the taxonomy of attacks in IoT environments [25]. Understanding challenges in regard to the security of IoT environments is an important factor in the defence against attacks like the denial of service attack. Sufian Hamee et al. sets apart the different challenges in IoT environments [13] while Chintan Patel et al. highlights the challenges in IoT regarding smart cities, models and applications [28]. A denial of service attack has to be detected before defensive measures can be taken. Elike Hodo et al. uses an artificial neural network to detect threats in an IoT environment [15]. Andria Procopiou et al. developed "ForChaos" which detects denial of service attacks using forecasting and chaos theory [29]. How to mitigate denial of service attacks is an important factor. Shaker Alanazi et al. shows how a mesh routing protocol can be used against denial of service attacks [2]. Furthermore, Daniel Jeswin Nallathambi et al. use honeypots to mitigate denial of service attacks in IoT environments [4]. A blockchain mitigation solution is presented by Minhaj Ahmad Khan et al. [21]. Both detection and mitigation of cyber IoT attacks in the smart grid environment are demonstrated by Yasin Yılmaz et al. [36].

Our research provides a high level IoT architectural model where the main driving factor is to prevent liability claims from external parties. Furthermore, the model is flexible, allowing other solutions to incorporate whereas most other presented solutions focus on a specific technical problem or part of the network with little regard for the surrounding business processes which are equally important. The model is used to shape an IoT infrastructure in such a way that readily available detection and mitigation strategies can be implemented rather than a specific technical solution.

4 Model

In this section we present the IoT architecture model to detect, prevent and diminish the contribution to DDoS attacks. Moreover, we discuss the different IoT architecture layers and define some new elements pertaining to DoS prevention, asset management, patch management and network monitoring.

4.1 Restrictions

The model in this paper focuses on networks that can potentially contribute to DoS attacks. Therefore, we do not consider internal only networks or air gapped networks since the traffic can never enter the internet and cause harm to others. There is a multitude of different protocols that IoT devices can use, each with their own positive and negative properties. Solutions applied in one specific protocol potentially does not necessary work in another protocol. Therefore, we only focus on IP networks and non IP networks.

4.2 IoT layered architecture

The model we present here is based on the general IoT architecture as discussed in section 3.1.3. The gateway is the last point of defence before the traffic reaches the internet. Therefore, we focus on the network layer where the gateway resides and the lower object layer. Within these two layers we define 3 layers of defence. The network layer corresponds to the edge mitigation defence layer while the object layer can be further divided into two separated defensive layers namely; the intermediate mitigation layer and the object mitigation layer. The defensive layers within the objects and network IoT architecture layers is depicted in Figure 2.

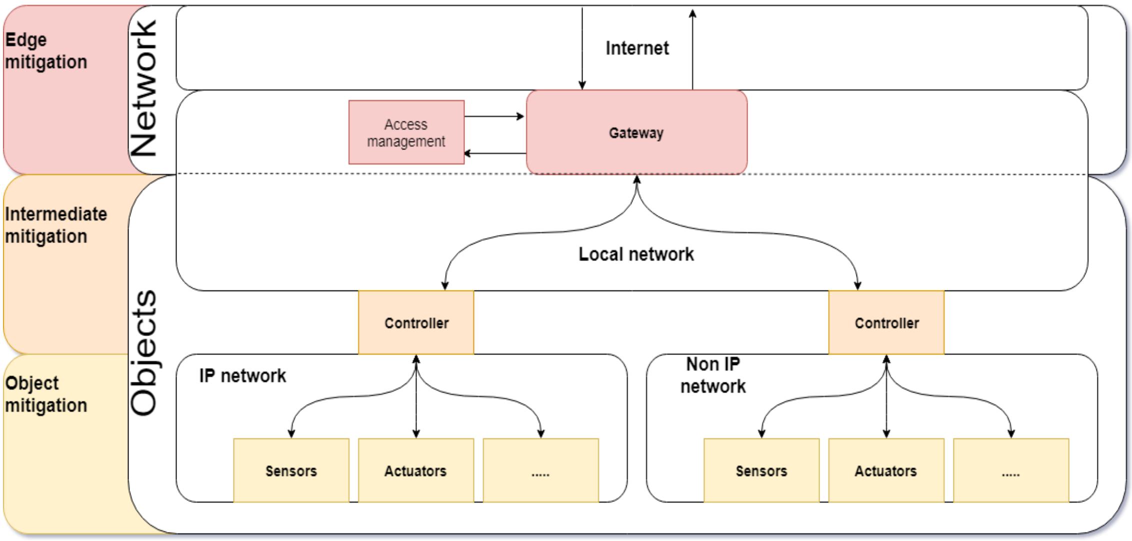

Figure 2: Defensive layers.

Edge mitigation is achieved by utilizing the gateway from the network layer while the intermediate mitigation layer achieves mitigation by utilizing the IoT controllers. The object mitigation layer focuses on the IoT devices themselves, though this layer is not within our scope as discussed in section 4.1. Ideally mitigation happens on the lowest layer possible, in our case the intermediate mitigation layer. The IoT architecture objects, network, support and application layers including the 3 mitigation layers are depicted in Figure 3.

4.3 Module overview

The model is built around the mitigation layers as specified in section 4.2. Specifically the model is built around the edge mitigation layer utilizing the gateway and intermediate mitigation layer utilizing the IoT controllers. To achieve successful mitigation we need more elements than the IoT controllers and gateway. Therefore, we specify new elements pertaining to network monitoring, detection, mitigation strategies, reporting and asset management. The model itself consists of 8 elements or modules two of which are the IoT controllers and the gateway. Therefore, we specify 6 new modules namely;

1. (D)DoS detection module (DDM);
2. Control module (CM);
3. Mitigation decision module (MDM);
4. Update module (UM);
5. Reporting module (RM);
6. Asset management module (AMM).

A high level overview of the modules is given in Figure 4. The DDM is the starting point of the model. The DDM is responsible for detection of (D)DoS attacks originating from the IoT devices going through the gateway. The DDM is connected with the CM where the necessary information is passed to the RM for archiving purposes or sent to the MDM to implement mitigation on a detected (D)DoS attack originating from the IoT network. Furthermore, the CM is responsible to assign a threat level to the reported attack. The MDM decides what mitigation strategy is implemented. The MDM implements access control lists on the gateway based on the logged layer 2 and layer 3 routing information or sends it to the UM for device updates. The UM updates the corresponding IoT controllers with new versions of firmware, software or configuration updates. All important statistics and routing information is stored at the RM where it is stored in a central database. The RM sends information needed at other modules from the central database.

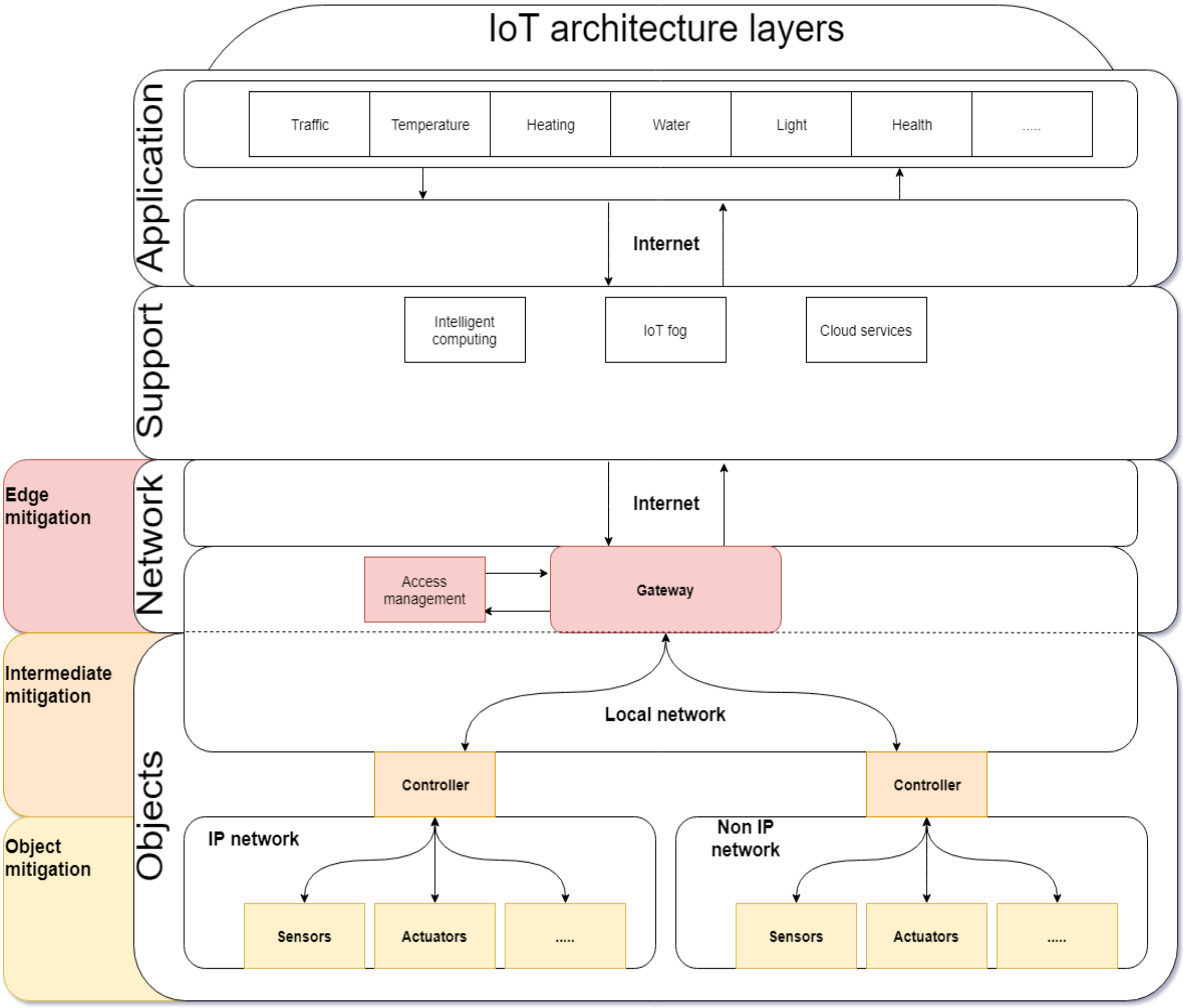

Figure 3: IoT architecture.

The AMM is responsible for maintenance of the IoT controllers and to verify they conform business policies. In the coming sub chapters we discuss each module in more detail.

4.3.1 DDoS detection module (DDM)

The DDM is responsible for detecting possible (D)DoS attacks originating from the IoT network and reporting this to the CM. The first step is to feed the traffic passing through the gateway to the DMM. Ideally all the traffic is passed through the DDM for detection, though in high speed networks this might not be possible. Therefore, samples of the traffic can still be used to detect (D)DoS attacks with the DDM. Sampling instead of the full traffic flow lowers the detection rate of the DDM. (D)DoS attacks often have a prolonged duration especially with ongoing attacks. Therefore, sampling is a viable solution that can be utilized to detect (D)DoS attacks. The DDM detects (D)DoS attacks with 3 possible methods namely; signature based, anomaly based or threshold based as depicted in Figure 5.

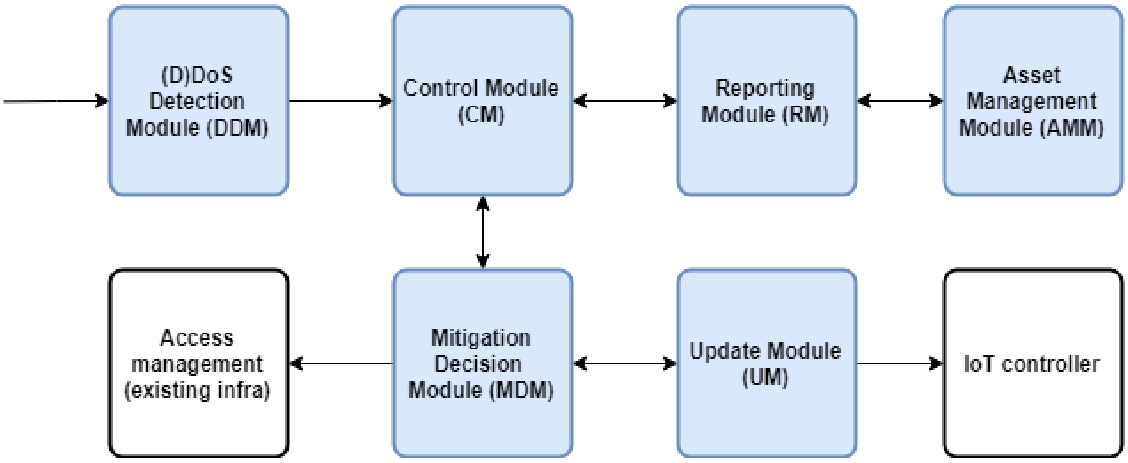
(D)DoS Detection Module (DDM)
Control Module (CM)
Reporting Module (RM)
Asset Management Module (AMM)
Access management (existing infra)
Mitigation Decision Module (MDM)
Update Module (UM)
IoT controller

Figure 4: Overview modules.

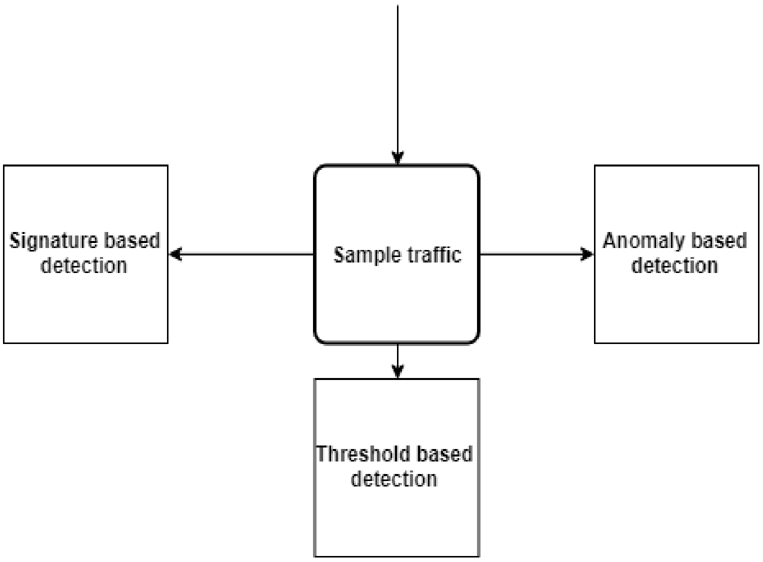

Figure 5: DDM Sample traffic.

Anomaly based detection uses a baseline behaviour database in which the DDM compares the analysed traffic flow with the behaviour database. It is forwarded to the statistic collector if the analysed traffic does not fall within the normal behaviour. The analysed traffic is added to the baseline behaviour database if the traffic falls within the current baseline behaviour database. The result is that the baseline behaviour database keeps expanding with new data after the initial learning phase is over. Depending on the size of the IoT infrastructure the normal behaviour database resides inside the device that facilitates the DDM functionality. Furthermore, for larger IoT infrastructures the normal behaviour database is clustered allowing the DDM to scale to large size IoT environments. The anomaly logic is depicted in Figure 6.

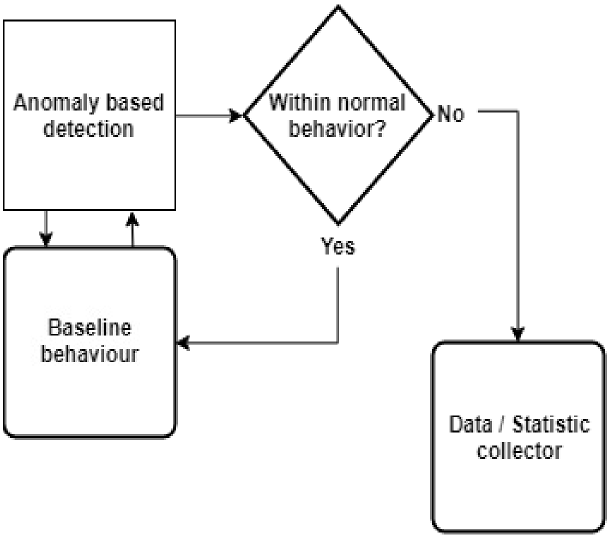

Figure 6: DDM anomaly based detection.

Threshold based detection uses the available metrics from the network equipment in the current IT infrastructure. In our case we specified packet length, packets per second and bandwidth. Though, these can be expanded if other metrics are available in the network which, largely depends on the gateway and IoT controllers used. Traffic is sent to the baseline behaviour database if no threshold is exceeded. However, it is forwarded to the statistic collector if a threshold is exceeded. The threshold based detection logic is depicted in Figure 7.

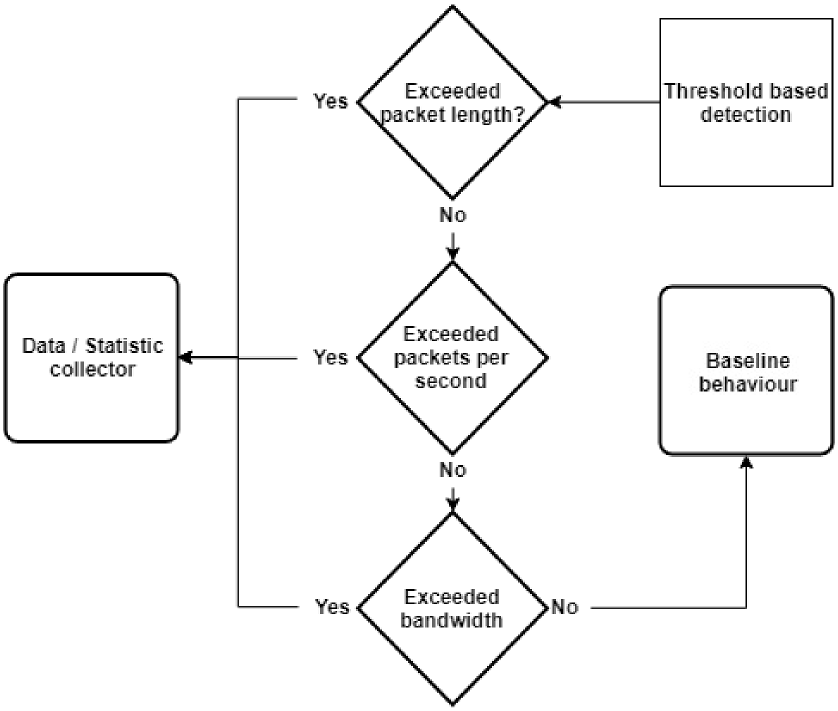
Threshold based detection
Exceeded packet length?
Yes
No
Exceeded packets per second
Yes
No
Exceeded bandwidth
Yes
No
Data / Statistic collector
Baseline behaviour

Figure 7: DDM threshold based detection.

Signature based detection uses a database where known attack signatures are stored. The analysed traffic is compared to the known signatures. The traffic is forwarded to the baseline behaviour database if no match with a signature is found. Otherwise it is forwarded to the statistic collector. The signature logic is depicted in Figure 8.

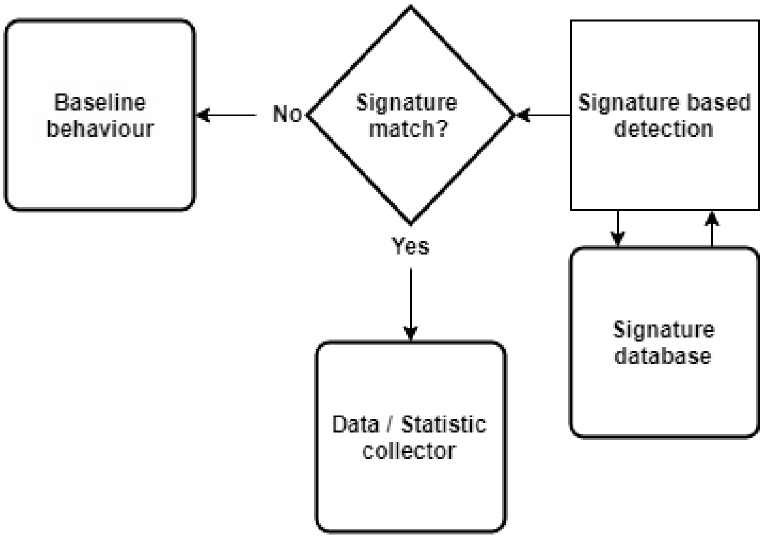

Figure 8: DDM signature based detection.

The statistic collector is responsible for setting the relevant attack ID based on the detection method used (anomaly, signature or threshold) and what triggered the attack. This attack ID is unique for every (D)DoS attack, in turn a mitigation strategy is coupled with the attack ID. Furthermore, the statistic collector retrieves the layer 2 and layer 3 routing information from which the access control lists are created. A timestamp is also added to correlate attacks and for archiving purposes. All detected (D)DoS attacks from anomaly, signature and threshold based detection come together at the statistic collector. This is the ideal place to extract possible new signatures from the detected (D)DoS attacks. The newly extracted attack signatures together with signatures downloaded from the internet are inserted into the signature database. The statistic collector logic is depicted in Figure 9.

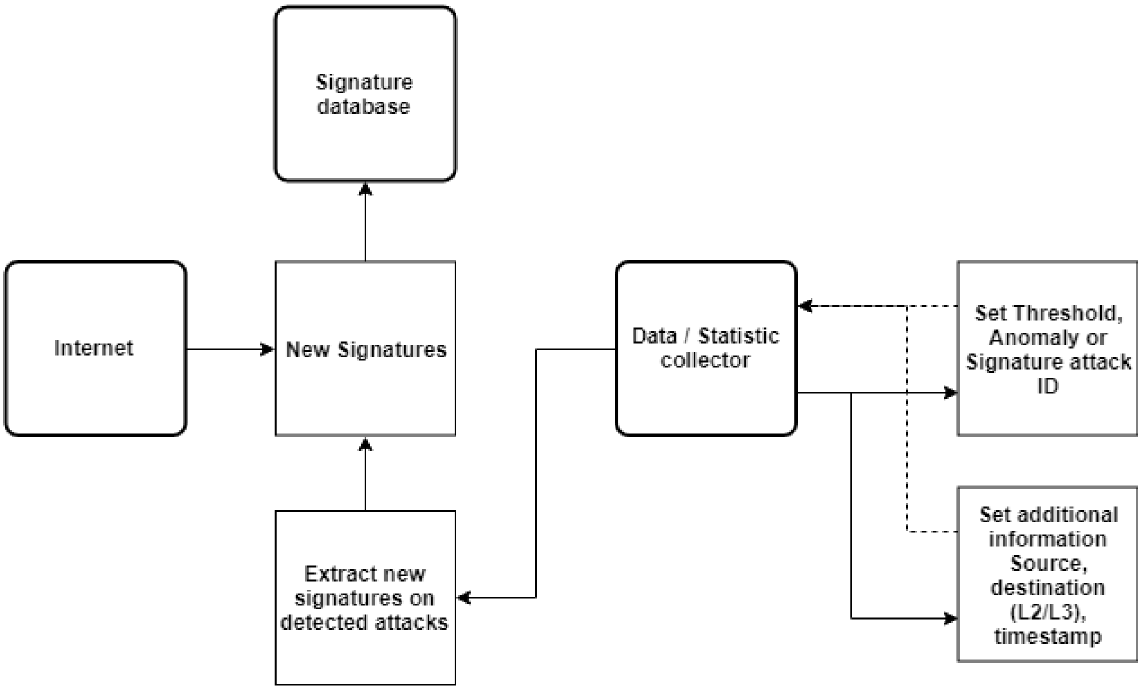
Signature database
Internet
New Signatures
Data / Statistic collector
Set Threshold, Anomaly or Signature attack ID
Extract new signatures on detected attacks
Set additional information Source, destination (L2/L3), timestamp

Figure 9: DDM statistics collector.

The final step of the DDM is to send the information from the statistic collector to the CM. The complete DDM logic is depicted in Figure 11.

4.3.2 Control module (CM)

The CM's main task is to set the threat level of the detected (D)DoS attack. The information from the DDM first arrives in the statistic extractor from the CM. The statistic extractor sends the timestamp, attack ID (anomaly, signature or threshold) and source, destination layer 2 and layer 3 information to the threat analyser. The timestamp, attack ID, layer 2/3 and threat level are also send to the RM for archiving. The statistic extractor logic is depicted in Figure 10.

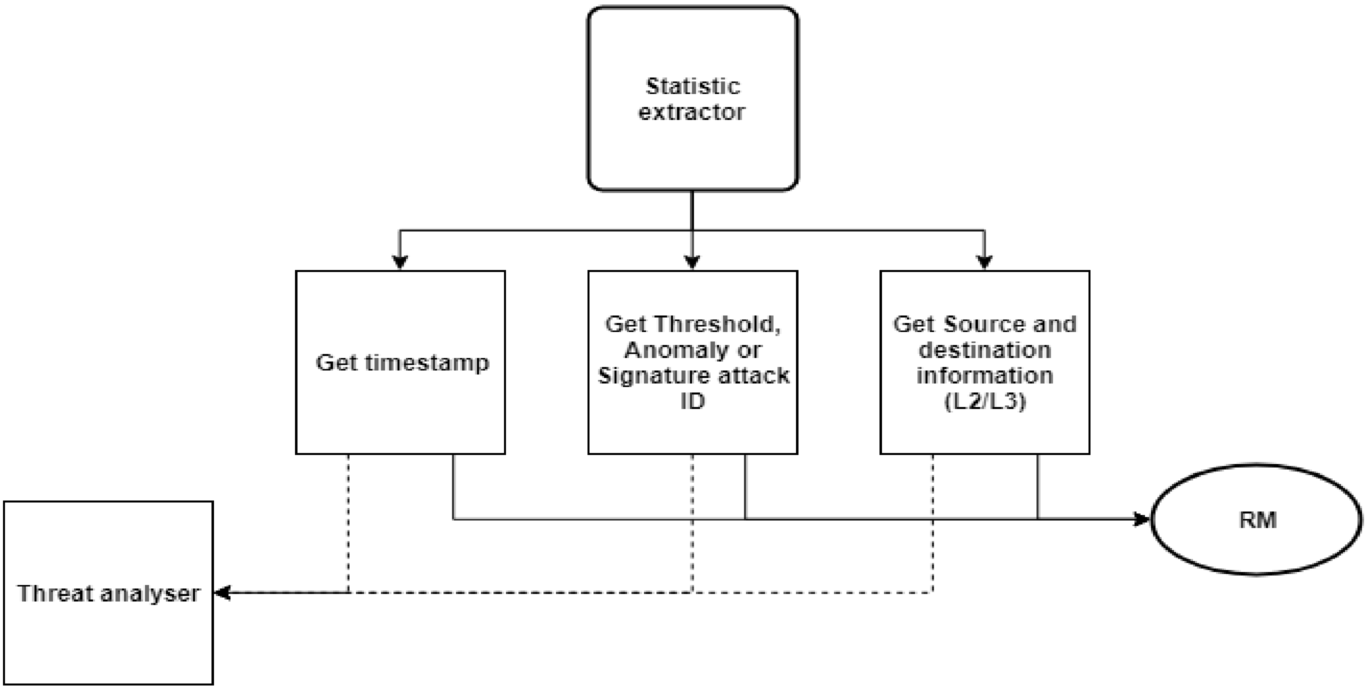
Statistic extractor
Get timestamp
Get Threshold, Anomaly or Signature attack ID
Get Source and destination information (L2/L3)
Threat analyser
RM

Figure 10: CM statistics extractor.

The threat analyser queries the threat level database to extract criteria to determine the threat level of the detected (D)DoS attack. Criteria can include critical time period, affected network segment, or detected signatures. The next step is to correlate previous attack occurrences with the currently analysed (D)DoS attack. The RM is queried for previous occurrences pertaining to the same network segment, source, destination, time frame and attack ID. This data is then compared to the current (D)DoS attack. Decisions to increase the threat level can be made by correlating the available information from the RM. The CM threat analyser logic is depicted in Figure 12.

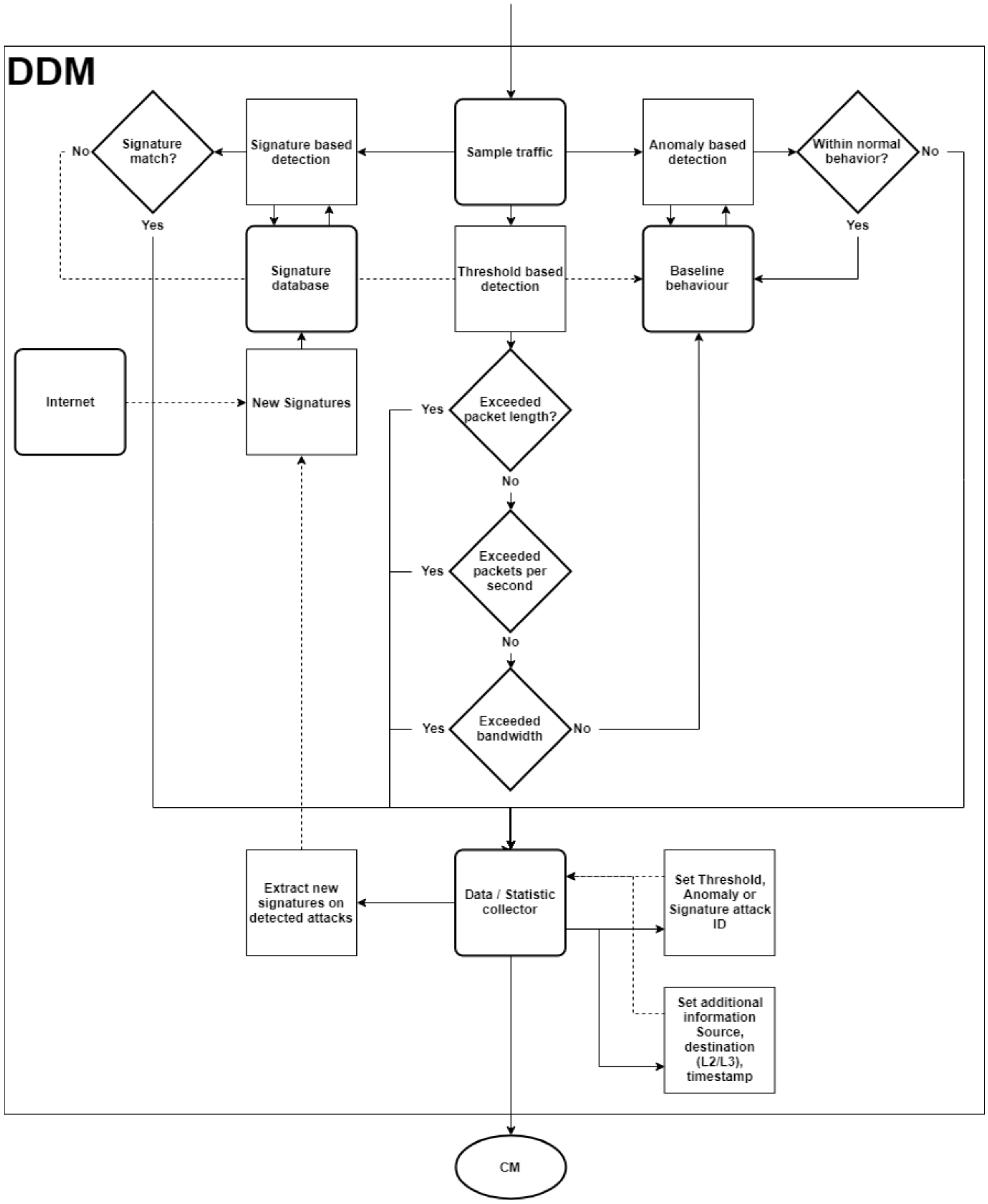
DDM
Sample traffic
Signature based detection
Signature match?
No
Yes
Anomaly based detection
Within normal behavior?
No
Yes
Signature database
Threshold based detection
Baseline behaviour
Internet
New Signatures
Exceeded packet length?
Yes
No
Exceeded packets per second
Yes
No
Exceeded bandwidth
Yes
No
Extract new signatures on detected attacks
Data / Statistic collector
Set Threshold, Anomaly or Signature attack ID
Set additional information Source, destination (L2/L3), timestamp
CM

Figure 11: DDM (DDoS detection module).

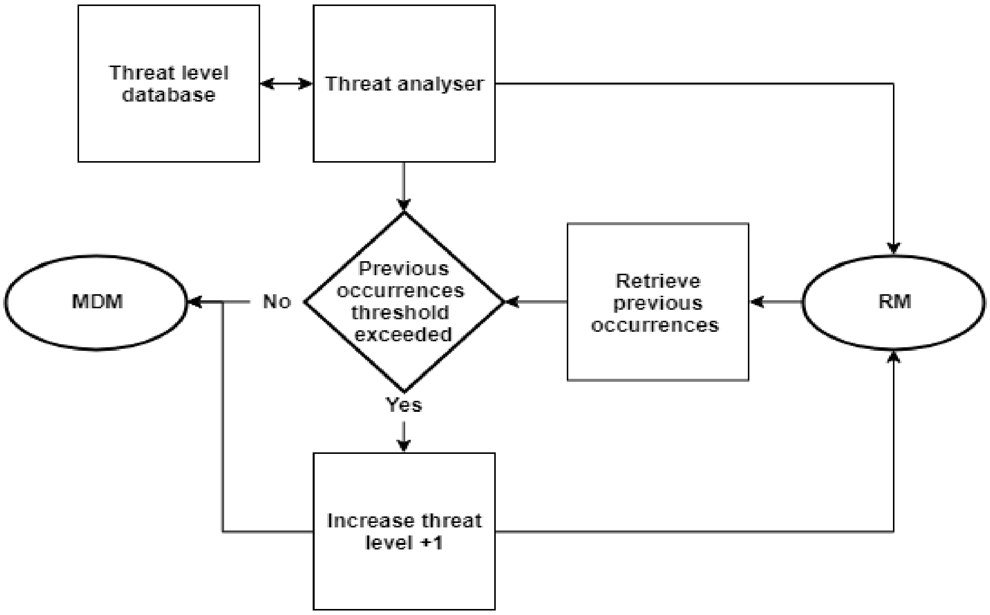

Figure 12: CM threat analyser.

For example the threat level is increased if the same (D)DoS attack is registered within a set time frame and the previous mitigation solution did not alleviate the problem. The threat level is increased if the same IoT network segment is sending out (D)DoS traffic within a set time frame or a set threshold is exceeded. Other possible uses are extracted from the information stored in the RM. Furthermore, the CM passes the information from the MDM and UM to the RM. More specifically it sends the IoT configuration ID and the error report from the UM and the implemented gateway or IoT controller mitigation solution from the MDM to the RM as depicted in Figure 13. The complete CM overview is depicted in Figure 14.

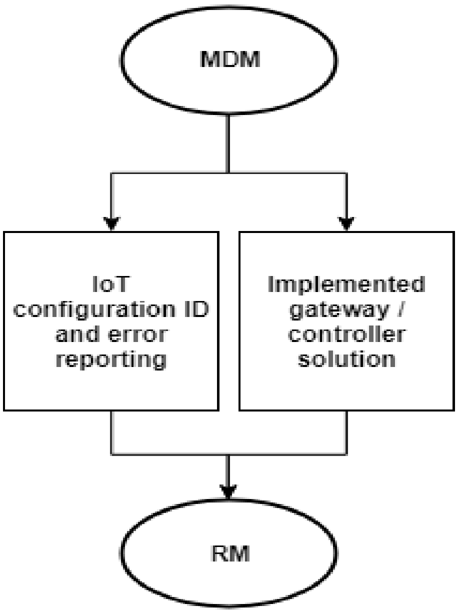
MDM
IoT configuration ID and error reporting
Implemented gateway / controller solution
RM

Figure 13: CM reporting of other modules information.

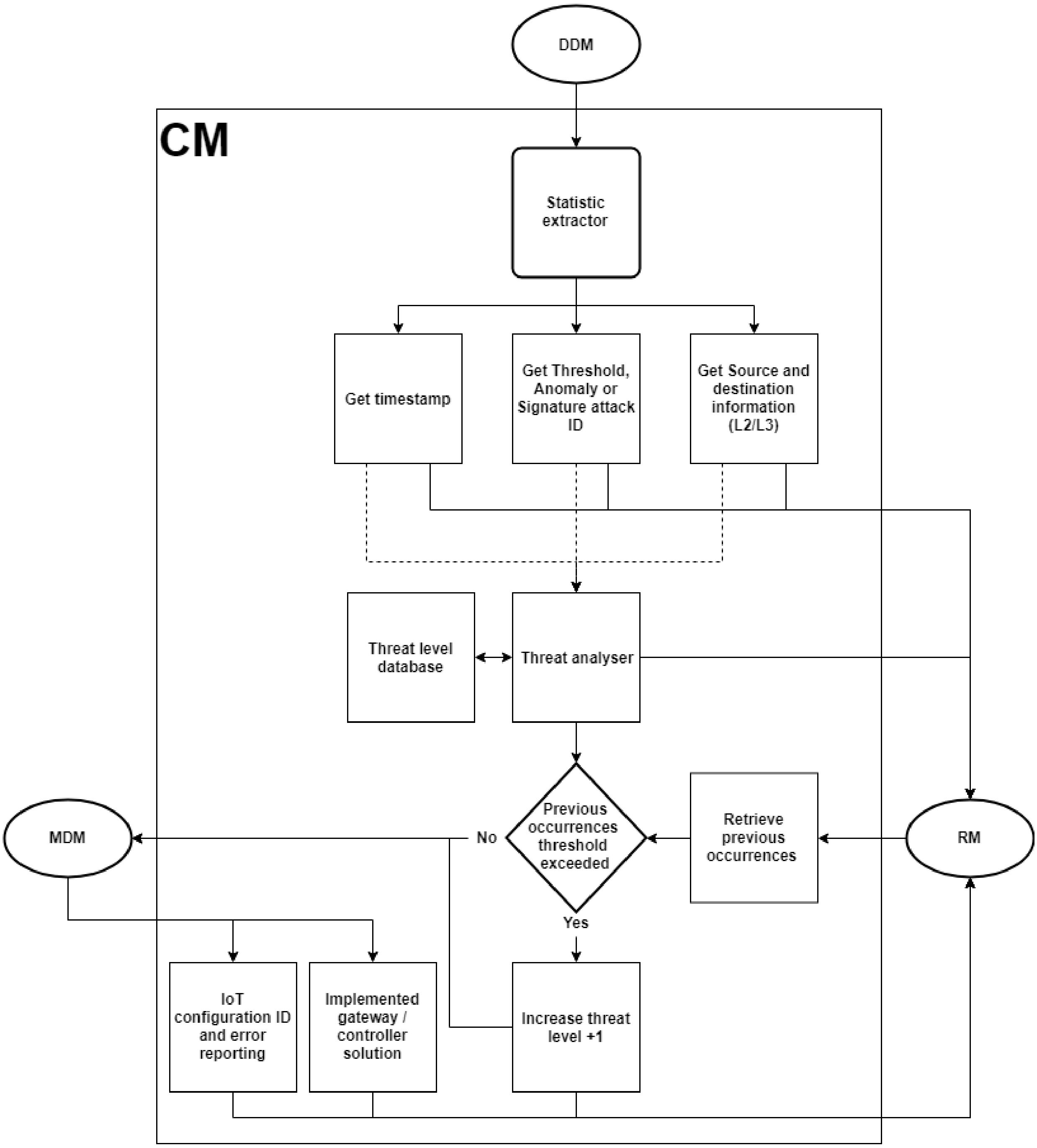
DDM
CM
Statistic extractor
Get timestamp
Get Threshold, Anomaly or Signature attack ID
Get Source and destination information (L2/L3)
Threat level database
Threat analyser
MDM
No
Previous occurrences threshold exceeded
Retrieve previous occurrences
RM
Yes
IoT configuration ID and error reporting
Implemented gateway / controller solution
Increase threat level +1

Figure 14: CM (Control module).

4.3.3 Mitigation decision module (MDM)

The main task of the MDM is to determine where the mitigation takes place. In our case this is the edge mitigation layer and the intermediate mitigation layer. The MDM first checks if the threat level has an emergency level. The emergency threat level is used to push an emergency access control list to the gateway in case of severe (D)DoS attacks. The emergency logic is depicted in Figure 15.

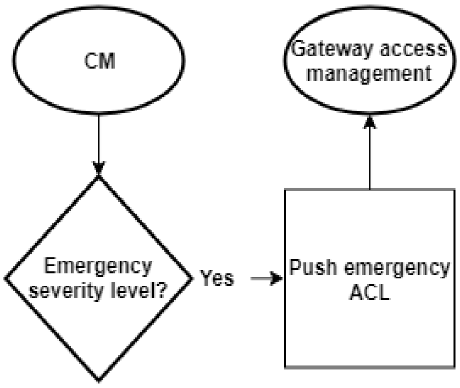

Figure 15: MDM emergency check.

In our case the emergency threat level is based on the total bandwidth usage versus the normal behaviour baseline measurement. The attack is stopped at the gateway to prevent IoT equipment from participating in a (D)DoS attack. Likewise, the destination addresses are used to determine if traffic is being sent to places where it should not go. The emergency severity level is activated if the threshold for unknown destinations is exceeded. Next the MDM checks if the IoT controllers involved in the attack are push-able. The term "push-able" refers to the IoT controller accepting firmware, software, configuration or access control lists pushed from the UM. Therefore, intermediate mitigation is not possible and edge mitigation is used instead. An access control list based on the IoT controller layer 3 information is pushed to the gateway in the event that an IoT controller does not support push-able updates. The push-able IoT controller logic is depicted in Figure 16.

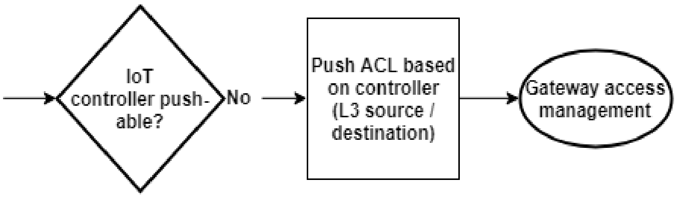
IoT controller push-able?
No
Push ACL based on controller (L3 source / destination)
Gateway access management

Figure 16: MDM IoT controller update check.

Next the MDM checks if the reported threat level falls within the edge mitigation strategy or the intermediate mitigation strategy. An access control list based on the layer 3 traffic is pushed to the gateway if the threat level falls within the edge mitigation strategy. The intermediate mitigation threat levels get sent to the controller mitigation. The edge threat check is depicted in Figure 17.

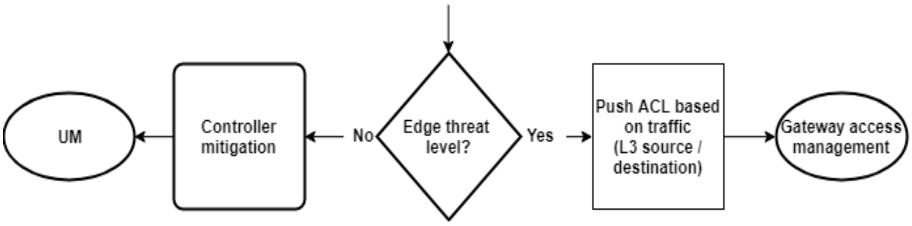
UM
Controller mitigation
No
Edge threat level?
Yes
Push ACL based on traffic (L3 source / destination)
Gateway access management

Figure 17: MDM edge threat check.

All access control lists implemented at the gateway also get sent to the CM before reaching the RM to be stored for later use. Likewise, the controller mitigation also sends an ID to the CM before reaching the RM to indicate that intermediate mitigation is implemented by the UM. The reporting of the implemented mitigation strategy logic is depicted in figure 18.

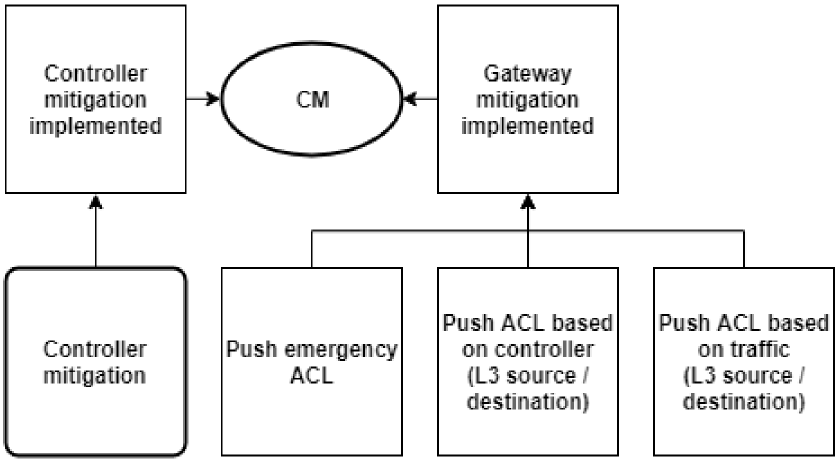
Controller mitigation implemented
CM
Gateway mitigation implemented
Controller mitigation
Push emergency ACL
Push ACL based on controller (L3 source / destination)
Push ACL based on traffic (L3 source / destination)

Figure 18: MDM mitigation strategy reporting.

Furthermore, the MDM reports the IoT configuration ID and the IoT error report from the UM to the CM before reaching the RM for archiving which is depicted in figure 19. The complete MDM overview is depicted in Figure 21.

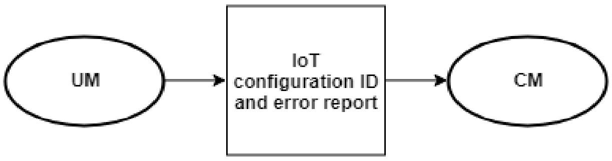

Figure 19: MDM reporting of UM originated data.

4.3.4 Update module (UM)

The UM is responsible for implementing the intermediate mitigation layer implementations. The 4 main tasks of the UM is to check for firmware, software, configuration and access control list updates. The UM first checks for new versions of firmware for the IoT controller in question. The new firmware version is pushed to the IoT controller and an updated IoT controller configuration ID is reported to the MDM who sends the configuration ID to the RM for storage. The firmware check logic is depicted in Figure 20.

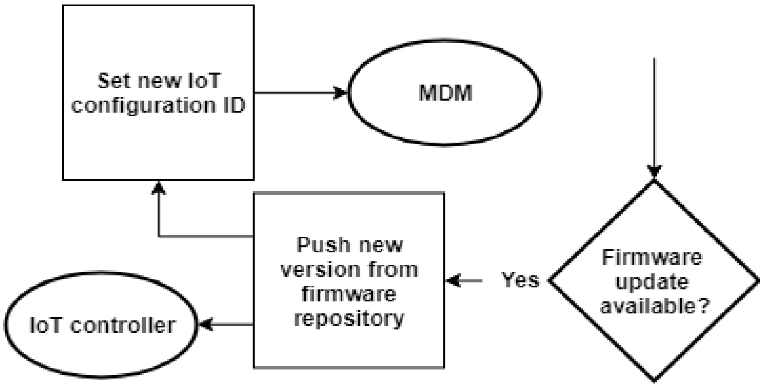
Set new IoT configuration ID
MDM
Push new version from firmware repository
Yes
Firmware update available?
IoT controller

Figure 20: UM firmware check.

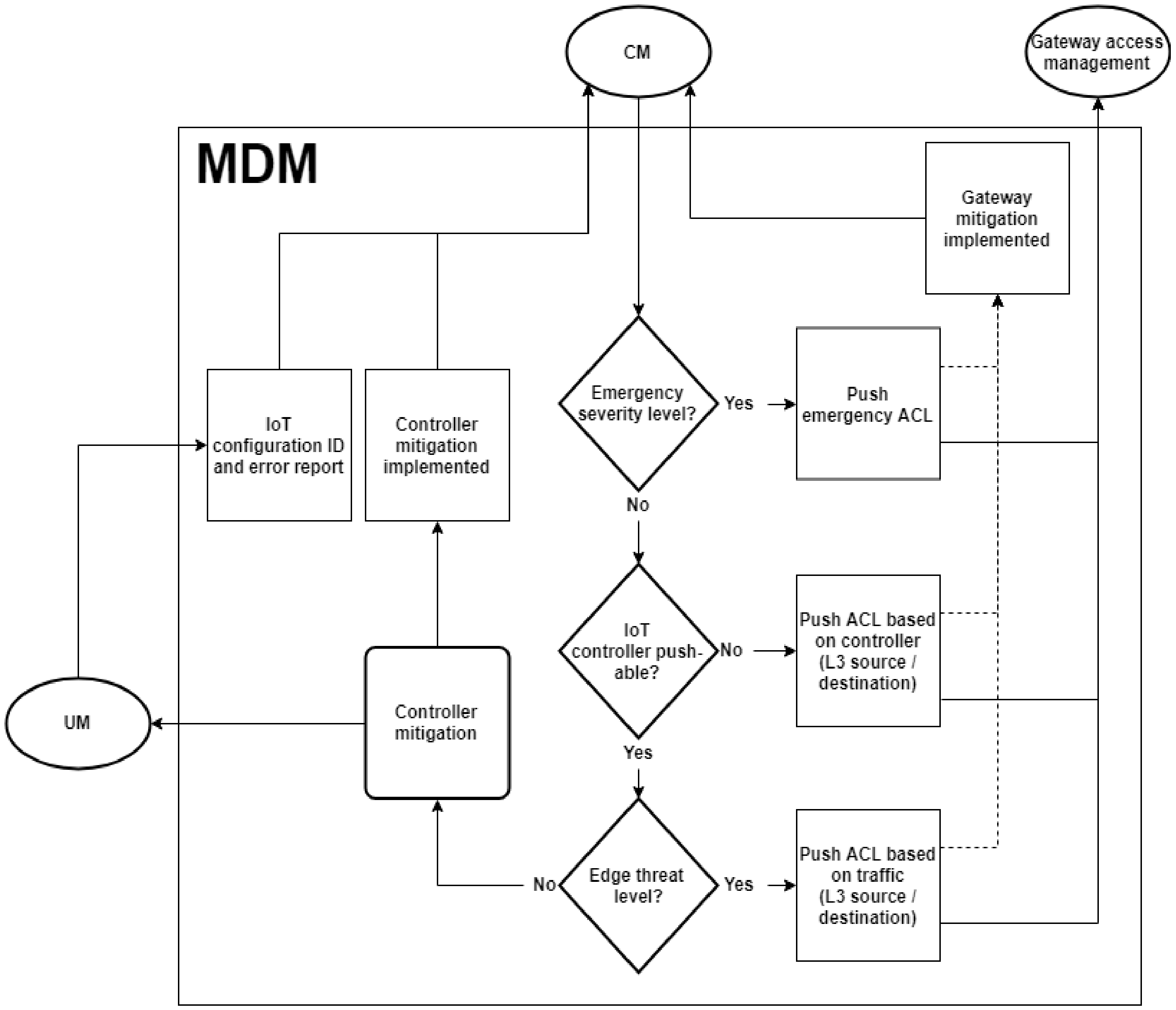
CM
Gateway access management
MDM
Gateway mitigation implemented
IoT configuration ID and error report
Controller mitigation implemented
Emergency severity level?
Yes
No
Push emergency ACL
IoT controller push-able?
No
Yes
Push ACL based on controller (L3 source / destination)
UM
Controller mitigation
No
Edge threat level?
Yes
Push ACL based on traffic (L3 source / destination)

Figure 21: MDM (Mitigation decision module).

Next the UM checks if new software is available for the IoT controller. Likewise, the new software version is pushed to the IoT controller and a new IoT configuration ID is reported back to the MDM. The software check logic is depicted in Figure 22.

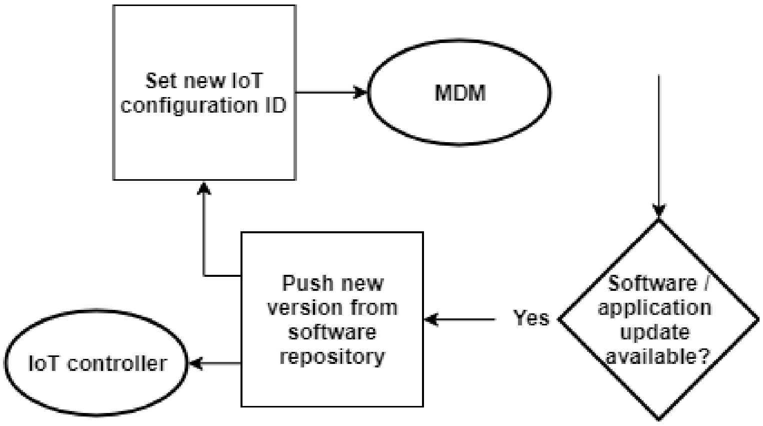
Set new IoT configuration ID
MDM
Push new version from software repository
Yes
Software / application update available?
IoT controller

Figure 22: UM software check.

Next the UM checks if the IoT controller has configuration updates available. The new configuration is pushed to the IoT controller if a new version is available from the repository. The configuration check logic is depicted in Figure 23.

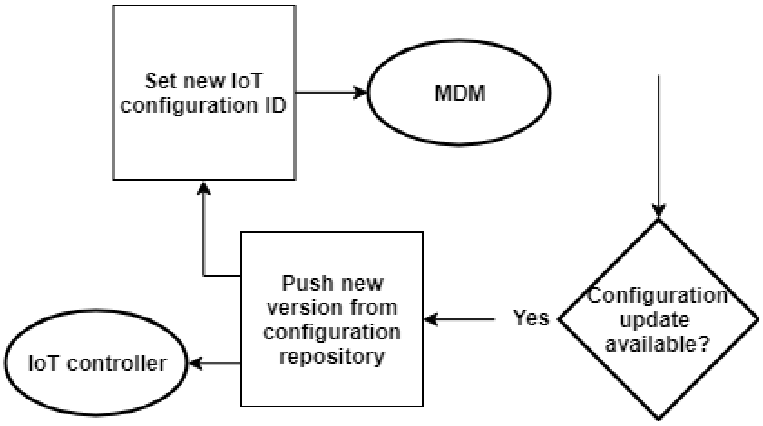
Set new IoT configuration ID
MDM
Push new version from configuration repository
Yes
Configuration update available?
IoT controller

Figure 23: UM configuration check.

Finally the UM checks for newly or updated access control lists for the IoT controller in question. An error report with the current IoT configuration ID is sent to the MDM if all four checks failed. The access control list check logic is depicted in Figure 24.

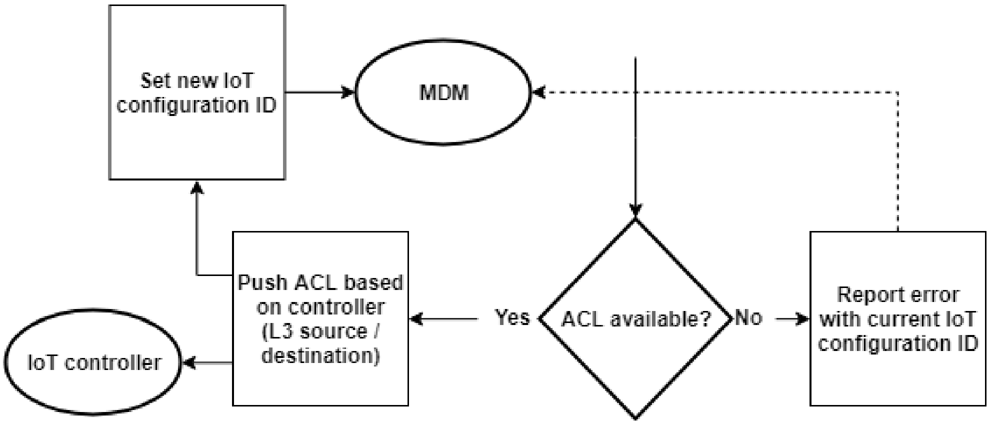
Set new IoT configuration ID
MDM
Push ACL based on controller (L3 source / destination)
Yes
ACL available?
No
Report error with current IoT configuration ID
IoT controller

Figure 24: UM access control list check.

This error report can in turn be used to determine if a previous implemented intermediate mitigation solution was tried. The CM then uses this information to increase the threat level so the next mitigation iteration takes place at the edge mitigation layer. Furthermore, the IoT controllers should indicate if a pushed firmware, software, configuration or access control list was successful or not. This feedback is important, else the UM does not know if the pushed update was successful and in turn the RM does not have the full information stored which is used by the CM to make threat level assessments. The complete UM overview is depicted in Figure 25.

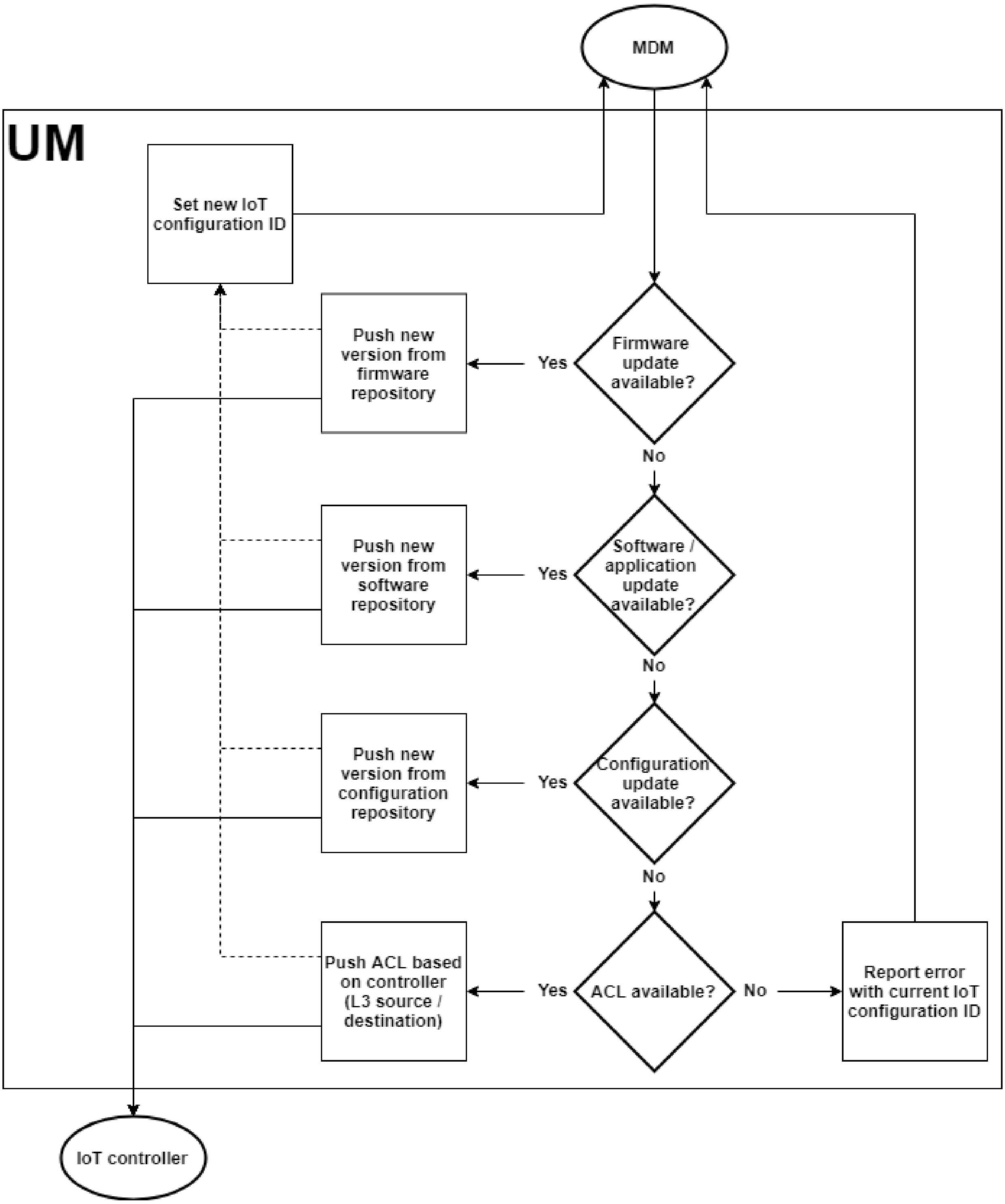
MDM
UM
Set new IoT configuration ID
Push new version from firmware repository
Yes
Firmware update available?
No
Push new version from software repository
Yes
Software / application update available?
No
Push new version from configuration repository
Yes
Configuration update available?
No
Push ACL based on controller (L3 source / destination)
Yes
ACL available?
No
Report error with current IoT configuration ID
IoT controller

Figure 25: UM (Update module).

4.3.5 Reporting module (RM)

The main task of the RM is to store all information generated by the other modules. The information stored at the RM is in turn used by the CM to make decisions about the threat level of a detected (D)DoS attack. The RM first extracts the data sent from the other modules. The RM statistic extractor logic is depicted in Figure 26.

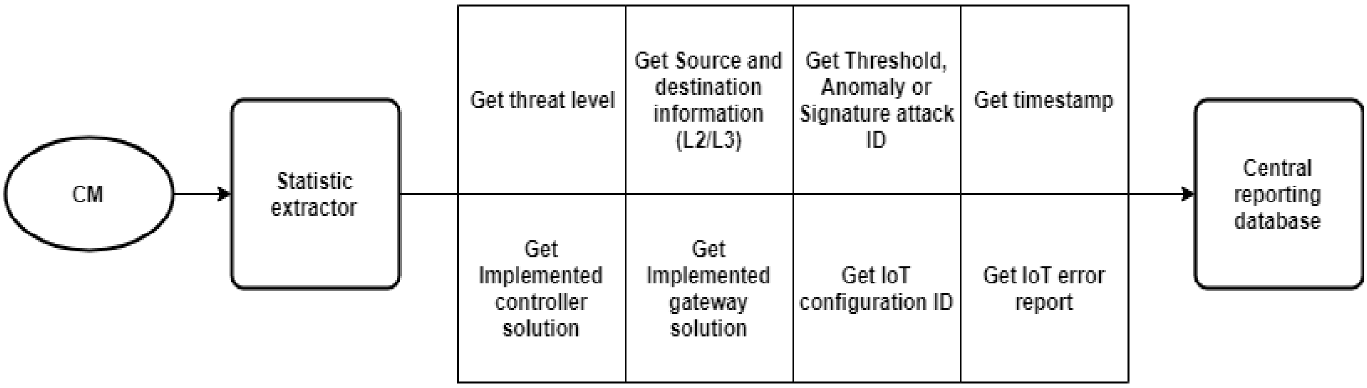
CM
Statistic extractor
Get threat level
Get Source and destination information (L2/L3)
Get Threshold, Anomaly or Signature attack ID
Get timestamp
Get Implemented controller solution
Get Implemented gateway solution
Get IoT configuration ID
Get IoT error report
Central reporting database

Figure 26: RM statistic extractor.

More specifically it receives the source and destination layer 2 and layer 3 information, the attack ID and the timestamp from the DDM. The CM reports the threat level assigned after analysing the (D)DoS attack. Furthermore, the MDM reports the implemented mitigation solution of the IoT controller or the gateway. The UM is responsible for delivering the IoT configuration IDs and the IoT error reports generated. Lastly, the AMM is responsible for populating the central reporting database with the maintenance IDs of the IoT controllers. The RM maintenance logic is depicted in Figure 27, while Table 4.3.5 shows the different information streams and where they originate and pass-through.

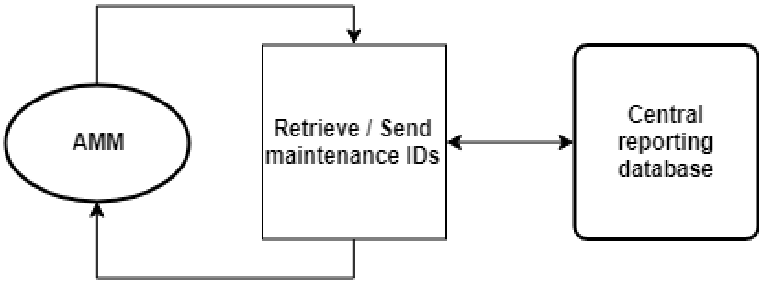

Figure 27: RM maintenance id.

Data	Originate from	Passes through	Stored at
Threat level	CM	-	RM
Source and destination information (L2/L3)	DDM	CM	RM
Threshold, Anomaly or Signature attack ID	DDM	CM	RM
Timestamp	DDM	CM	RM
Implemented controller solution	MDM	CM	RM
Implemented gateway solution	MDM	CM	RM
IoT configuration ID	UM	MDM, CM	RM
IoT error report	UM	MDM, CM	RM
Maintenance IDs	AAM	-	RM

Table 1: Model information distribution

The information stored at the RM is the central point from where new detection criteria are formed. New metrics that are of value to formulate new detection criteria or threat level assignments are added to the central reporting database in the RM. In turn the RM offers this information to the relevant modules. The complete RM overview is depicted in Figure 28.

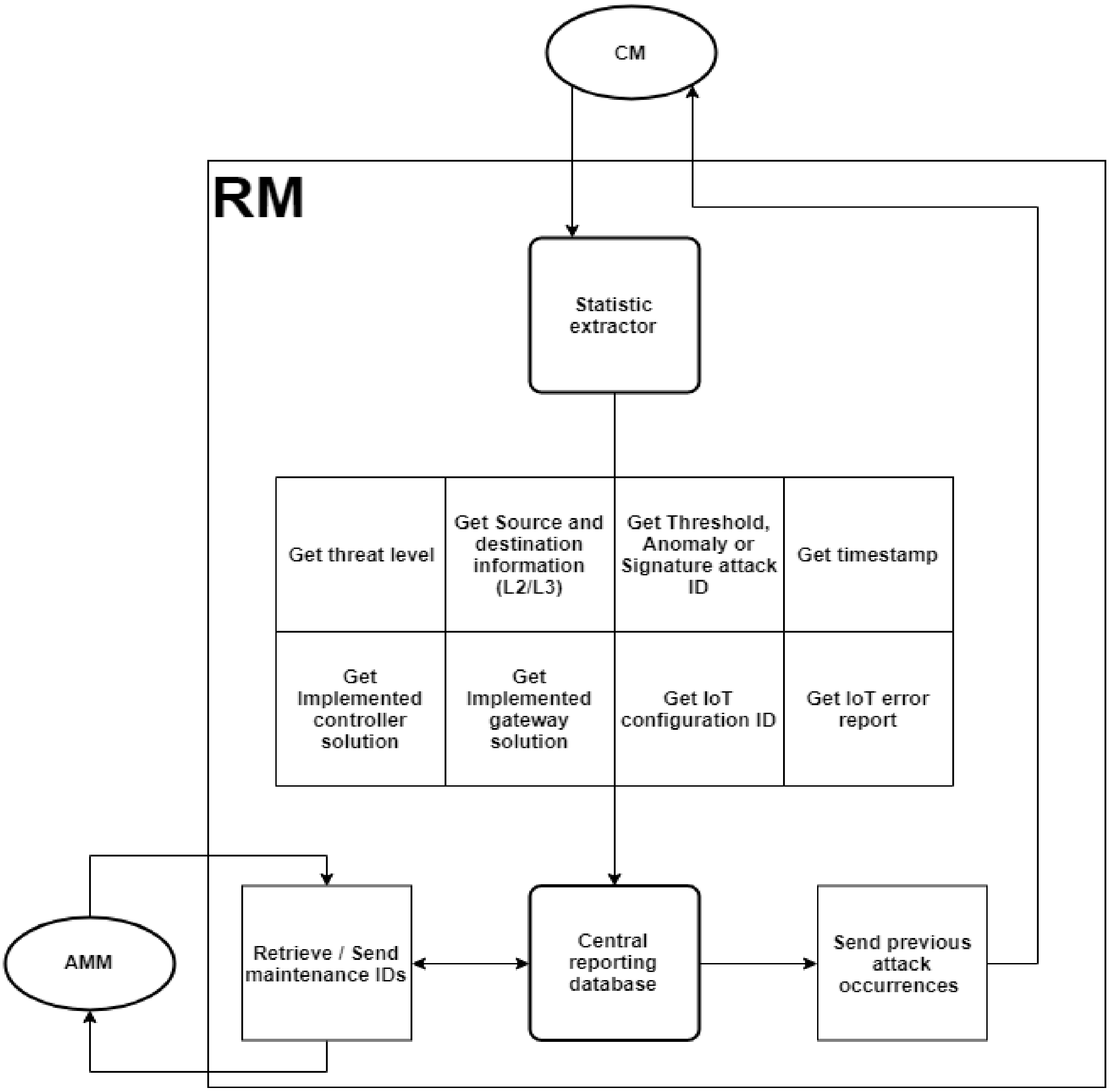
CM
RM
Statistic extractor
Get threat level
Get Source and destination information (L2/L3)
Get Threshold, Anomaly or Signature attack ID
Get timestamp
Get Implemented controller solution
Get Implemented gateway solution
Get IoT configuration ID
Get IoT error report
AMM
Retrieve / Send maintenance IDs
Central reporting database
Send previous attack occurrences

Figure 28: RM (Reporting module).

4.3.6 Asset management module (AMM)

The AMM is responsible for the maintenance and decommission of new and existing IoT controllers. New IoT controllers come in through the manufacturer where the device identity is established. Next the manufacturer distributes the IoT controllers to their customers. At the delivery stage the IoT controller arrives at the company who then establishes an owner. The owner is responsible for the IoT controller in question. This creates a central point for security issues, information and regulatory policies. At the deployment stage the IoT controller is supplied with the initial configuration along with access control and provisioning. The initial delivery and preparation of the IoT controllers is depicted in Figure 29.

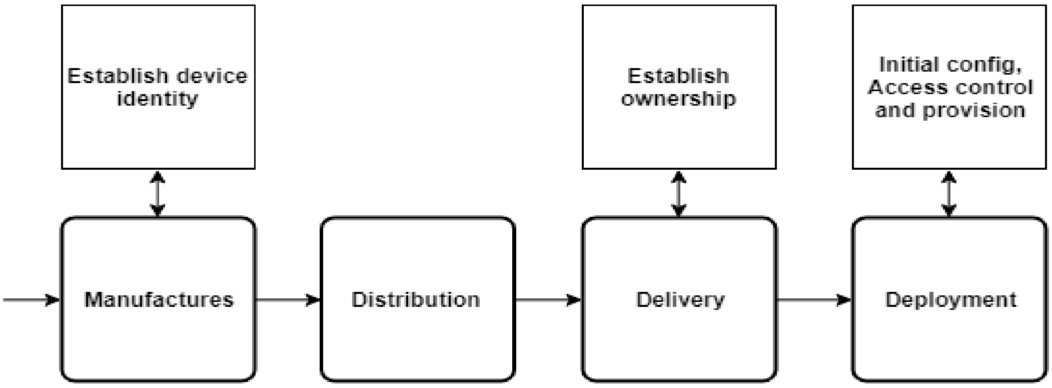
Establish device identity
Establish ownership
Initial config, Access control and provision
Manufactures
Distribution
Delivery
Deployment

Figure 29: AMM initial delivery.

The RM is queried for existing IoT controllers that are scheduled for maintenance. The previous maintenance ID is retrieved which contains the maintenance history of the IoT controller in question. A new initial configuration along with access control and provisioning are provided if the IoT controller did not receive maintenance before. After the IoT controller is provisioned the business policies are checked to validate if the IoT controller adheres to the policies. The IoT controller is re-deployed if the business policies are satisfied. Otherwise, the IoT controller is decommissioned and the ownership is removed. The maintenance check logic is depicted in Figure 30.

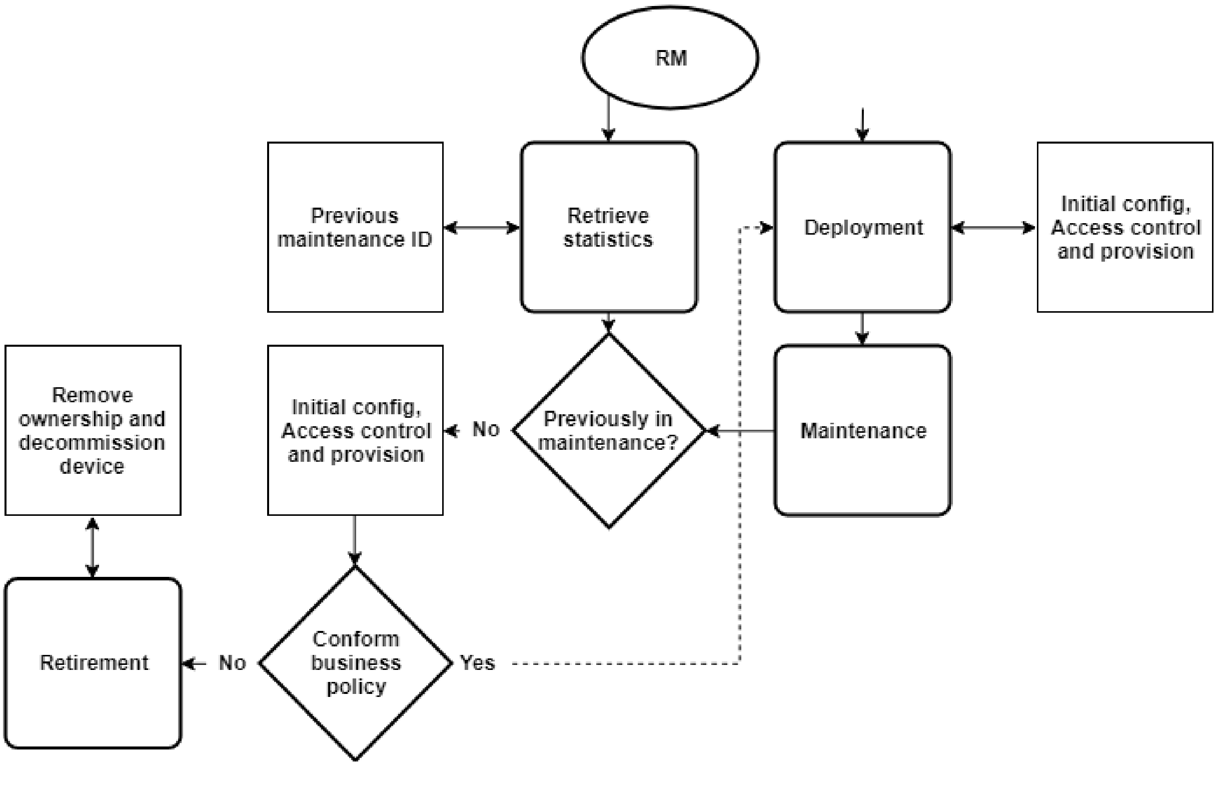
RM
Previous maintenance ID
Retrieve statistics
Deployment
Initial config, Access control and provision
Remove ownership and decommission device
Initial config, Access control and provision
No
Previously in maintenance?
Maintenance
Retirement
No
Conform business policy
Yes

Figure 30: AMM previously in maintenance.

The error is checked for IoT controllers that have been in maintenance before. IoT controllers with a different error than before are treated as normal. They get a new initial configuration with access control and provisioning and are checked against the business policies before being re-deployed. The same error check logic is depicted in Figure 31.

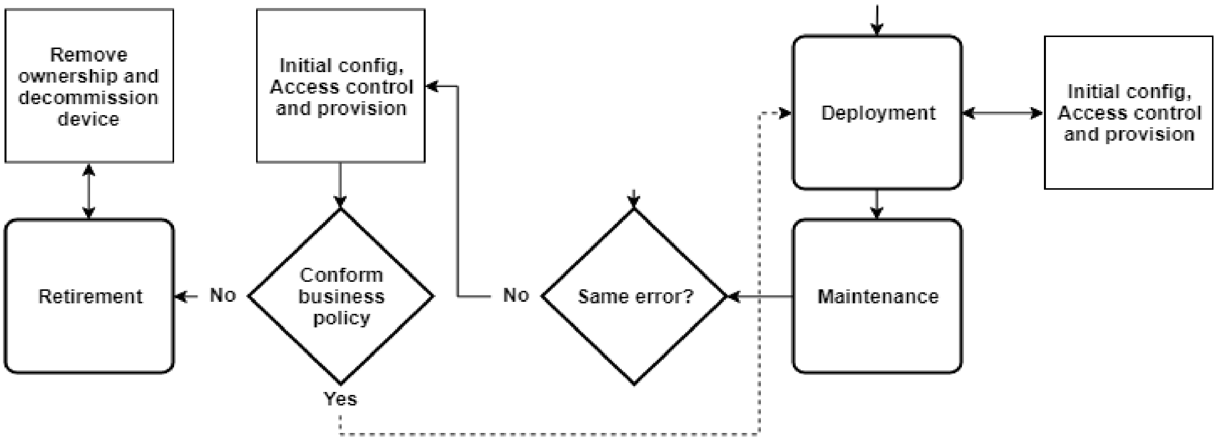
Remove ownership and decommission device
Initial config, Access control and provision
Deployment
Initial config, Access control and provision
Retirement
No
Conform business policy
No
Same error?
Maintenance
Yes

Figure 31: AMM same error check.

The threshold is checked for IoT controllers that have had the same error before. IoT controllers where the threshold is not exceeded are treated as normal. They get a new initial configuration with access control and provisioning and are checked against the business policies before being re-deployed. Otherwise, the IoT controller in question might be breaking down or experiences malfunctioning. Therefore, a new possible solution has to be found to alleviate the failing IoT controller which often is aided by the manufacturer of the IoT controller in question. The IoT controller is decommissioned if there is no solution available. Otherwise, they get a new initial configuration with access control and provisioning and are checked against the business policies before being re-deployed. The AMM new solution for the IoT controller is depicted in Figure 32.

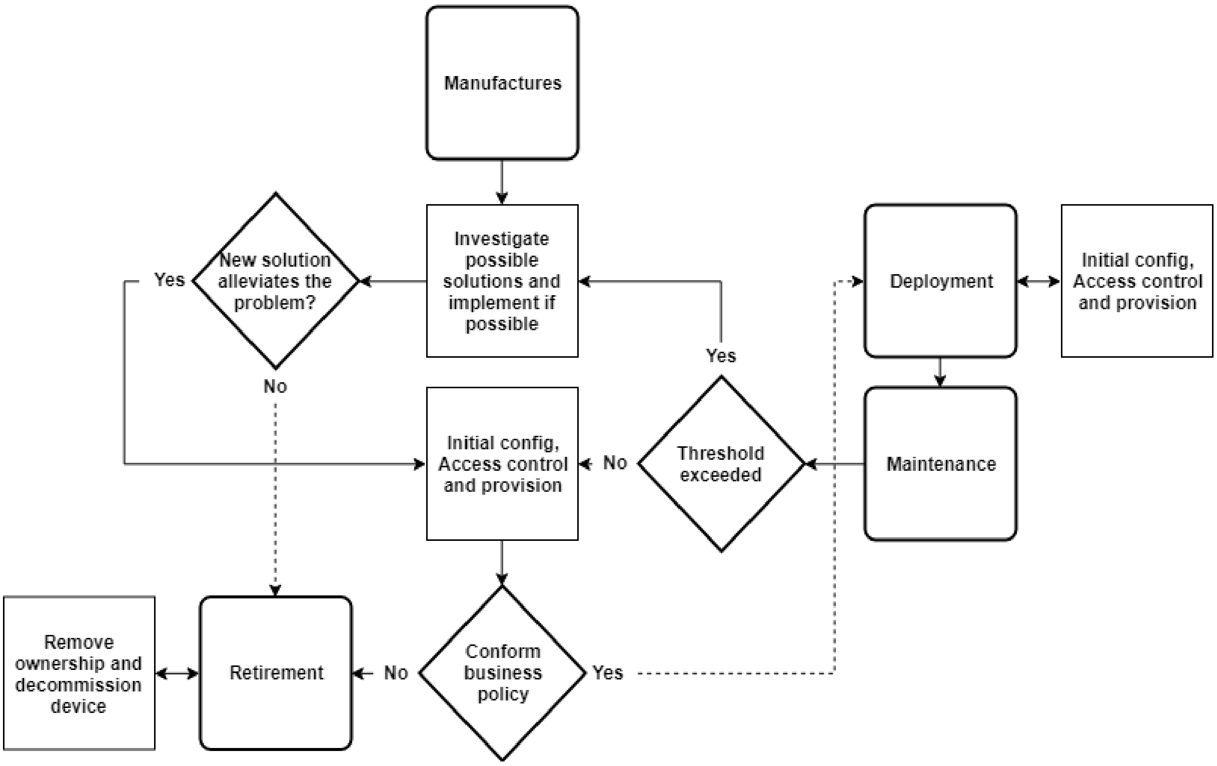
Manufactures
Yes
New solution alleviates the problem?
Investigate possible solutions and implement if possible
Deployment
Initial config, Access control and provision
Yes
No
Initial config, Access control and provision
No
Threshold exceeded
Maintenance
Remove ownership and decommission device
Retirement
No
Conform business policy
Yes

Figure 32: AMM new solution logic.

Furthermore, the maintenance IDs are sent to the RM for storage and later re-use within the AMM module, which is depicted in Figure 33. The complete AMM overview is depicted in Figure 34.

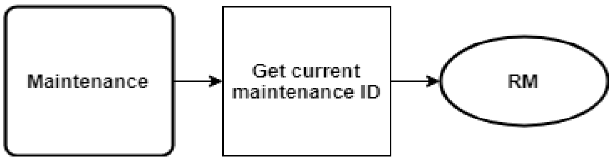

Figure 33: AMM maintenance ID reporting to RM.

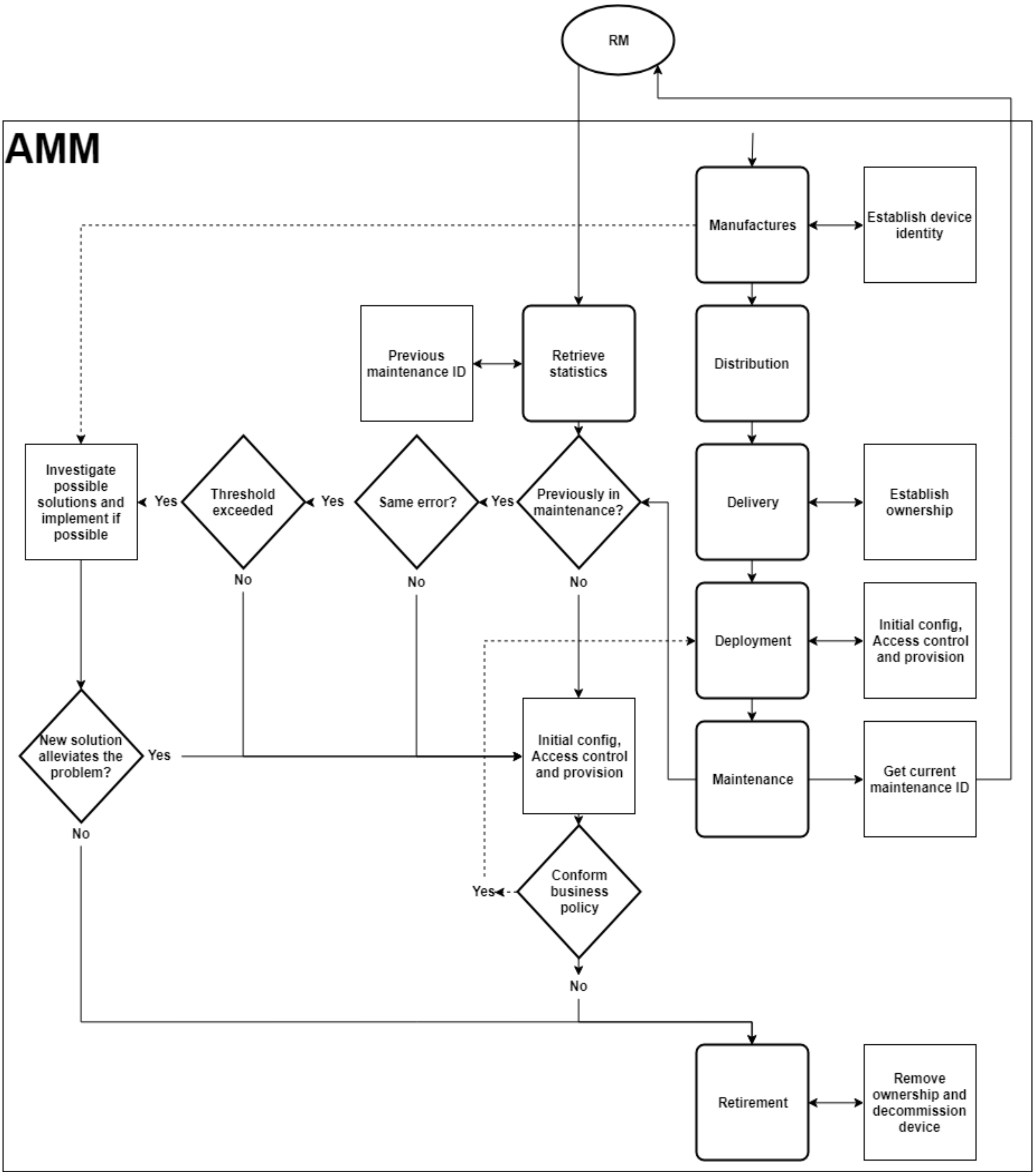
RM
AMM
Manufactures
Establish device identity
Previous maintenance ID
Retrieve statistics
Distribution
Investigate possible solutions and implement if possible
Yes
Threshold exceeded
Yes
Same error?
Yes
Previously in maintenance?
Delivery
Establish ownership
No
No
No
Deployment
Initial config, Access control and provision
New solution alleviates the problem?
Yes
Initial config, Access control and provision
Maintenance
Get current maintenance ID
No
Yes
Conform business policy
No
Retirement
Remove ownership and decommission device

Figure 34: AMM (Asset management module).

4.4 IoT architecture with modules

In this sub section we discuss the IoT architecture with the DDM, CM, MDM, UM, RM and AMM added to the infrastructure. The IoT architecture with added modules is depicted in Figure 35. The DDM and the UM are placed in the network layer of the IoT architecture. The DDM is connected to the gateway to enable sampling of the traffic flowing through the gateway. The network and support layers in the IoT architecture are usually connected through the internet in large scale IoT networks. Therefore, the DDM and UM reside inside the network layer. It would be unfeasible to send large amounts of traffic from the gateway to the DDM if the DDM resides in the support layer. Likewise, the firmware, software, configuration and access control repositories can grow to large volumes in large scale IoT networks. Therefore, the UM also resides inside the network layer. An advantage is that the UM and DDM only need the software and detection logic for a single site and not the whole IoT infrastructure as a whole. The CM, MDM, RM and AMM are placed in the support layer since these 4 modules are the core of the model where the decisions are made. This separation makes the model scale to large IoT infrastructures since multiple DDM modules can be connected to the CM. Likewise, the MDM can direct multiple UM modules at different sites. The MDM itself only updates access control lists on the gateway. Therefore, it should be no problem sending these over the internet to the gateway in question.

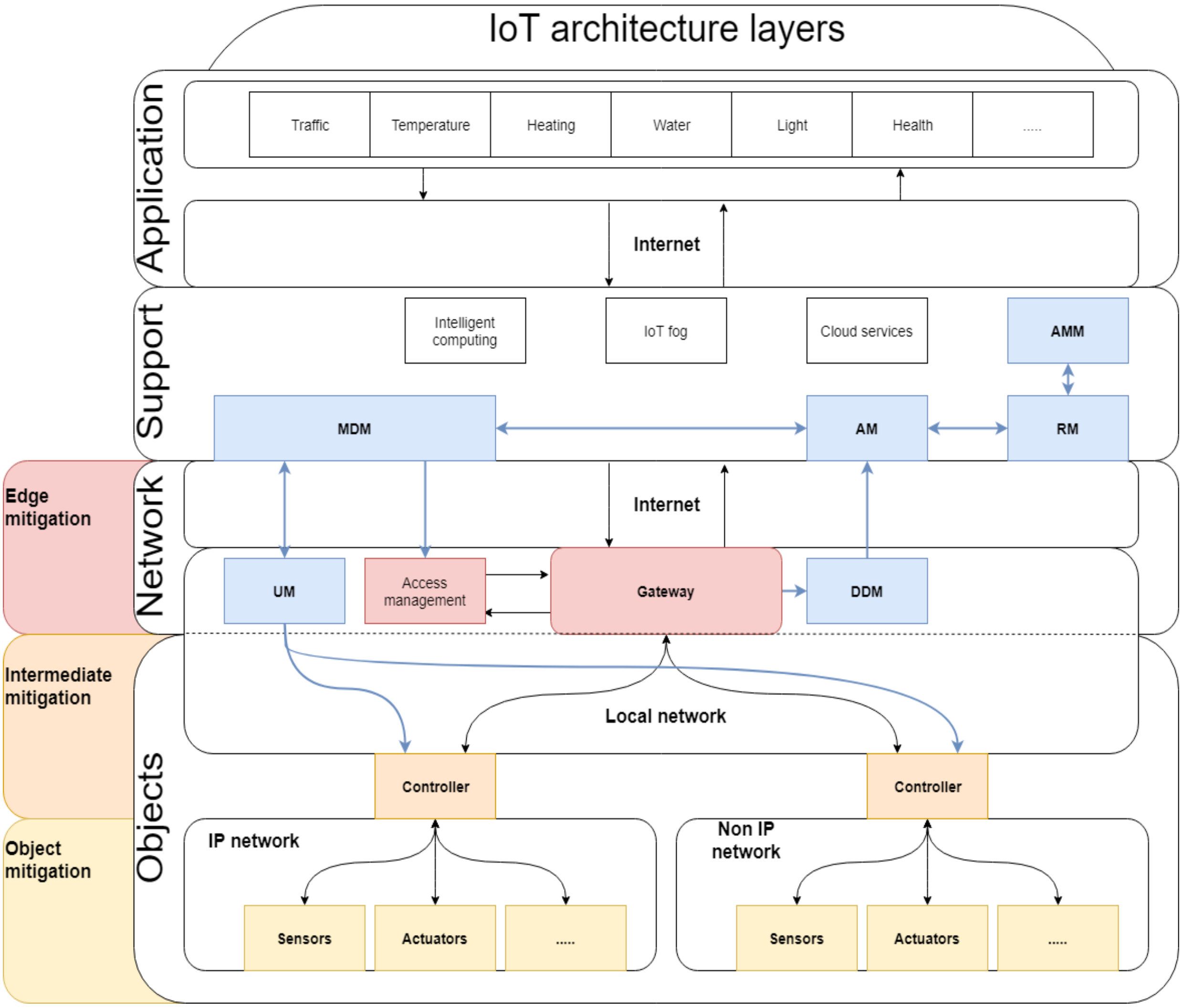

Figure 35: IoT architecture with modules added.

5 Discussion

The model presented in this paper is based on a general IoT architecture as discussed in section 3.1.3. The IoT architecture is a general conception of how IoT networks look and behave with common devices like the gateway and IoT controllers in place. Other types of IoT networks which do not use the same layout may not be able to use the model presented in this paper. In some cases the model can be adapted depending on how much the architecture in question differs from the used IoT architecture in this paper.

Our model defines the object, intermediate and edge defensive layers within the IoT architecture. Ideally mitigation is achieved at the lowest possible layer namely, the objects defensive layer. However, our model only utilises the edge and intermediate defensive layers. We feel the current standardisation for IoT in general is lacking in essential areas. There is a lack of lightweight encryption algorithms for low power IoT devices preventing secure transmission of data in a standardised way. Furthermore, the lack of protocol standards evolved into a multitude of different protocols and authentication mechanisms each with their own attributes and challenges existing on different layers. Therefore, we felt it was unfeasible to delve into the objects defensive mitigation layer presented in this paper but instead focus on the network parts where the traffic is translated to the internet protocol (IP).

The six modules described in this paper do not necessarily translate to devices though this is a possibility. At a minimum two devices are needed to keep the separation by layers intact. A possible setup is where the UM and DDM are combined into a single device while the CM, MDM, RM and AMM are combined into another device. With this setup the separation between the network layer and the support layer are kept intact. Small IoT networks can choose to implement all logic into a single device though the model loses the scalability property by removing the separation by layers.

The DDM in our model utilises anomaly, threshold and signature based detection methods. Other options like artificial intelligence (AI) or machine learning exists that might prove useful. However, we focused on well-known techniques rather then experimental new approaches. Likewise, mitigation techniques we use in our model have a high chance to be present in a network because they are based on IP networking techniques generally used in networking. However, the access control list mitigation strategies can have unintended side effects. Legitimate traffic is blocked if the implemented access control includes IP ranges of legitimate IoT devices. This is hard to counter in the event of a large scale (D)Dos attack originating from the IoT network with access control lists. Broad IP ranges are used to prevent the traffic from reaching the internet resulting in legitimate traffic being blocked. However, more sophisticated approaches like software defined networking, network function virtualisation, voting mechanisms or reputation based systems could be utilised to counter this. Likewise, our threshold attributes are based on layer 3 information. However, higher OSI layer information from the session and application layer give more granular attributes on which the thresholds could be defined. This gives more sophisticated triggers to detect (D)DoS attacks originating from the IoT network.

Our model is focused on preventing internal (D)DoS traffic from reaching the external internet. However, our model mitigation strategies could be affected by influences from the internet or higher layers from the IoT architecture. Network congestion can severely affect the mitigation strategies if the gateways and IoT controllers are reachable over the internet. An external (D)DoS attack targeted at the gateways or IoT controllers can prevent the mitigation strategies used in our model from being executed on the devices. Though, contribution to a (D)DoS attack is only minimal at best if the reach-ability is hindered by congestion. Gateways and IoT controllers that are not reachable over the internet can still be compromised if the higher support or application layer are compromised.

6 Conclusion

In this paper we present a model for IoT architectures, which can be utilised to prevent (D)DoS traffic originating internally from reaching the internet. Thereby, preventing possible liability claims. The model uses an IoT architecture separated by an object, network ,support and application layer. Our model specifies an objects, intermediate and edge defensive layer where mitigation strategies are executed. The three defensive layers are found within the objects and network layer of the IoT architecture. When (D)DoS traffic is detected the model tries to achieve mitigation at the lowest possible defensive layer. The lowest possible mitigation with our model is achieved in the intermediate defensive layer by utilising firmware, software, configuration or access control lists on the IoT controllers. Edge mitigation utilising access control lists is available if intermediate mitigation is not possible. Though, the chance of blocking legitimate traffic increases as layers upwards are traversed.

The model specifies 6 new modules that incorporate with the edge layer gateways and intermediate layer IoT controllers. The (D)doS detection module (DDM) is used to detect possible (D)DoS attacks by utilising anomaly, threshold or signature based detection on the gateway traffic. The control module is the threat analyser and the central point where information from other modules flows through. The mitigation decision module (MDM) implements access control list mitigation on the edge gateways and passes the intermediate mitigation strategies to the update module (UM). The UM is responsible for implementing the intermediate mitigation strategies on the IoT controllers. The asset management module (AMM) covers the IoT controllers deployed in the field and is responsible for business compliance and device decommissioning. The report module (RM) handles the storing of critical information used by the other modules to effectively assign threat levels and handle device decommissions.

7 Future Work

Our model describes the logic of the defined modules. An implementation or proof of concept is currently lacking. Measuring performance of the model is one area of interest. Specifically, the DDM detection methods, DDM traffic sampling rate, RM databases and CM threat logic are areas of interest in measuring performance. Furthermore, the scalability of the design is an area of interest in a future proof of concept as well as testing applicable hardware setups. For example, testing with different hardware but also testing the same hardware performing variations of the modules presented in this paper. An implementation or proof of concept can also uncover design flaws or possible optimisations of the design.

This paper delved into the edge and intermediate defensive layers of the IoT architecture. We omitted the object defensive layer as we felt standardisation and lightweight algorithms were lacking. However, incorporating the object defensive layer into the model will have positive effects as mitigation could be achieved at a lower layer than the current model allows. Furthermore, while mitigating at a lower layer the chance to block legitimate traffic decreases because mitigation is achieved closer to the source.

We did not specify a matrix for threat level assignments by the control module (CM) as this is largely depended on the metrics and attributes available in the network. Guidelines for threat level assignment could be created aiding in the implementation of the model into actual hardware. Though, a clear distinction should be made between threat levels assigned to the edge and intermediate defensive layers.

List of Tables

List of Figures

Abbreviations

Model abbreviations (chronological)

DDM	(D)DoS detection module
CM	Control module
MDM	Mitigation decision module
UM	Update module
RM	Reporting module
AMM	Asset management module

Other abbreviations (alphabetical)

ACL	Access Control List
AI	Artificial Intelligence
C&C	Command and Control
DDoS	Distributed Denial of Service
DNS	Domain Name System
DoS	Denial of Service
FCC	Federal Communications Commission
FFD	Full Function Device
FTC	Federal Trade Commission
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IP	Internet Protocol
ISP	Internet Service Provider
ML	Machine Learning
NEN	Nederlandse Norm
NFV	Network Function Virtualisation
OSI	Open Systems Interconnection
RFD	Reduced Function Device
SDN	Software Defined Networking
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

References

[1] Vipindev Adat and BB Gupta. “Security in Internet of Things: issues, challenges, taxonomy, and architecture”. In: *Telecommunication Systems* 67.3 (2018), pp. 423–441.

[2] Shaker Alanazi et al. “On resilience of Wireless Mesh routing protocol against DoS attacks in IoT-based ambient assisted living applications”. In: *2015 17th International Conference on E-health Networking, Application & Services (HealthCom)*. IEEE. 2015, pp. 205–210.

[3] Omar Alrawi et al. “Sok: Security evaluation of home-based iot deployments”. In: *Proceedings of the IEEE Symposium on Security and Privacy (S&P). https://doi. org/10.1109/SP*. 2019.

[4] M Anirudh, S Arul Thileeban, and Daniel Jeswin Nallathambi. “Use of honeypots for mitigating DoS attacks targeted on IoT networks”. In: *2017 International Conference on Computer, Communication and Signal Processing (ICCCSP)*. IEEE. 2017, pp. 1–4.

[5] Armir Bujari et al. “Standards, security and business models: key challenges for the IoT scenario”. In: *Mobile Networks and Applications* 23.1 (2018), pp. 147–154.

[6] Alibaba Clouder. *CERT Analysis on IoT Botnet and DDoS Attacks*. 2018 (accessed May 11, 2019). URL: `https://www.alibabacloud.com/blog/cert-analysis-on-iot-botnet-and-ddos-attacks_593859`.

[7] Cloudflare. *Famous DDoS Attacks | The Largest DDoS Attacks Of All Time*. 2018 (accessed May 12, 2019). URL: `https://www.cloudflare.com/learning/ddos/famous-ddos-attacks/`.

[8] Omar E Elejla et al. “Intrusion detection systems of ICMPv6-based DDoS attacks”. In: *Neural Computing and Applications* 30.1 (2018), pp. 45–56.

[9] enisa. *Major DDoS Attacks Involving IoT Devices*. 2016 (accessed May 11, 2019). URL: `https://www.enisa.europa.eu/publications/info-notes/major-ddos-attacks-involving-iot-devices`.

[10] Mario Frustaci et al. “Evaluating critical security issues of the IoT world: Present and Future challenges”. In: *IEEE Internet of Things Journal* 5.4 (2018), pp. 2483–2495.

[11] Antoine Gallais et al. “Denial-of-Sleep Attacks against IoT Networks”. In: *International Conference on Control, Decision and Information Technologies (CoDIT)*. 2019.

[12] Gartner. *Gartner Identifies Top 10 Strategic IoT Technologies and Trends*. 2018 (accessed May 13, 2019). URL: `hhttps://www.gartner.com/en/newsroom/press-releases/2018-11-07-gartner-identifies-top-10-strategic-iot-technologies-and-trends`.

[13] Sufian Hameed, Faraz Idris Khan, and Bilal Hameed. “Understanding Security Requirements and Challenges in Internet of Things (IoT): A Review”. In: *Journal of Computer Networks and Communications* 2019 (2019).

[14] *Het bericht 'Agentschap Telecom slaat alarm over hackbare apparaten'*. URL: `https://www.tweedekamer.nl/kamerstukken/kamervragen/detail?id=2018Z10731&did=2018D32722`.

[15] Elike Hodo et al. “Threat analysis of IoT networks using artificial neural network intrusion detection system”. In: *2016 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE. 2016, pp. 1–6.

[16] Fatima Hussain et al. “Machine Learning in IoT Security: Current Solutions and Future Challenges”. In: *arXiv preprint arXiv:1904.05735* (2019).

[17] IEEE. *IEEE 802.11-2016 - IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. 2016 (accessed May 15, 2019). URL: `https://standards.ieee.org/standard/802_11-2016.html`.

[18] IEEE. *IEEE 802.15.4-2015/Cor 1-2018 - IEEE Standard for Low-Rate Wireless Networks Corrigendum 1*. 2018 (accessed May 15, 2019). URL: `https://standards.ieee.org/standard/802_15_4-2015-Cor1-2018.html`.

[19] IEEE. *IEEE 802.3-2018 - IEEE Standard for Ethernet*. 2018 (accessed May 15, 2019). URL: `https://standards.ieee.org/standard/802_3-2018.html`.

[20] *Internet of Things heeft standaarden nodig*. Dec. 2016. URL: `https://smartindustry.nen.nl/internet-of-things-heeft-standaarden-nodig/`.

[21] Minhaj Ahmad Khan and Khaled Salah. "IoT security: Review, blockchain solutions, and open challenges". In: *Future Generation Computer Systems* 82 (2018), pp. 395–411.

[22] Vladimir Kuskov Mikhail Kuzin Yaroslav Shmelev. *New trends in the world of IoT threats*. 2018 (accessed May 12, 2019). URL: `https://securelist.com/new-trends-in-the-world-of-iot-threats/87991/`.

[23] Rashid Ali Mirza et al. "Preventing DDOS attacks from IOT". In: (2018).

[24] NaWas. *NaWas – National Scrubbing Center against DDoS attacks – a not-for-profit organization*. 2019 (accessed May 16, 2019). URL: `https://www.nbip.nl/en/nawas/`.

[25] Mukrimah Nawir et al. "Internet of Things (IoT): Taxonomy of security attacks". In: *2016 3rd International Conference on Electronic Design (ICED)*. IEEE. 2016, pp. 321–326.

[26] Alexander Gutnikov Oleg Kupreev Ekaterina Badovskaya. *DDoS Attacks in Q4 2018*. 2019 (accessed May 12, 2019). URL: `https://securelist.com/ddos-attacks-in-q4-2018/89565/`.

[27] Eric Osterweil, Angelos Stavrou, and Lixia Zhang. "20 Years of DDoS: a Call to Action". In: *arXiv preprint arXiv:1904.02739* (2019).

[28] Chintan Patel and Nishant Doshi. "Security Challenges in IoT Cyber World". In: *Security in Smart Cities: Models, Applications, and Challenges*. Springer, 2019, pp. 171–191.

[29] Andria Procopiou, Nikos Komninos, and Christos Douligeris. "ForChaos: Real Time Application DDoS Detection Using Forecasting and Chaos Theory in Smart Home IoT Network". In: *Wireless Communications and Mobile Computing* 2019 (2019).

[30] Gurubaran S. *Raise of IoT Botnets Responsible for Massive DDoS Attacks*. 2018 (accessed May 12, 2019). URL: `https://gbhackers.com/raise-of-iot-botnets-responsible-for-massive-ddos-attacks-q2-2018-threat-report/`.

[31] Karanbir Singh, Kanwalvir Singh Dhindsa, and Bharat Bhushan. "Threshold-based distributed DDoS attack detection in ISP networks". In: *Turkish Journal of Electrical Engineering & Computer Sciences* 26.4 (2018), pp. 1796–1811.

[32] Keshav Sinha and Partha Paul. "An Underground Mine Safety of Personnel's Using IoT". In: *Nanoelectronics, Circuits and Communication Systems*. Springer, 2019, pp. 77–88.

[33] Stamatia Triantopoulou. "An experimental analysis of current DDoS attacks based on a provider edge router honeynet". MA thesis. University of Piraeus, 2018.

[34] Vincent J. Vitkowsky. "The internet of things: A new era of cyber liability and insurance". In: (2015).

[35] Mark R. Warner. *Sen. Mark Warner Probes Friday;s Crippling Cyber Attack*. 2016 (accessed May 14, 2019). URL: `https://www.warner.senate.gov/public/index.cfm/pressreleases?ContentRecord_id=CD1BBB25-83E0-494D-B7E1-1C350A7CFCCA`.

[36] Yasin Yılmaz and Suleyman Uludag. "Timely Detection and Mitigation of IoT-based Cyberattacks in the Smart Grid". In: *Journal of the Franklin Institute* (2019).

[37] Ahmad Riza'ain Yusof, Nur Izura Udzir, and Ali Selamat. "Systematic literature review and taxonomy for DDoS attack detection and prediction". In: *International Journal of Digital Enterprise Technology* 1.3 (2019), pp. 292–315.