EMMA - EVOLVED MANAGEMENT AND MONITORING APPLICATION

Mihai BĂRBULESCU[1], Teodor IVĂNOAICA[2], Adrian RÂPĂ[3], Lucian PĂIUȘESCU[4], Mihai CARABAS[5]

Nowadays, infrastructure diversity is seen in every business, making hard for IT teams to debug and track all the issues arised. Evolved Management and Monitoring Assistant was designed based on the requirements of the Network Operation Center and Security Operation Center teams. Thus, we have provided a unified monitoring tool that adapts for multiple technologies, for multiple vendors, a tool that also reflects the flexibility of a team.

Keywords: emma, cdp, lldp, lacp.

1. Introduction

Managing the enterprise environment requires a multi-vendor monitoring platform for the infrastructure. The IT infrastructure, in every business, is comprised of network equipment, servers, desktop and mobile devices. In order to track the active devices and services and also find and debug all the problems that appears on such an IT infrastructure, one needs a tool that centralizes all the elements and is able to correlate among them. The main issue is the diversity of the equipment and services, being hard to find an application that is doing also tracking and active monitoring and debugging on the whole infrastructure. Two main elements in the business are the NOC (Network Operation Center) which takes care of the functionality of an IT infrastructure and the SOC (Security Operation Center) which takes care of the security issues in the business environment. In order to cover these aspects presented above, we propose EMMA (EVOLVED MANAGEMENT AND MONITORING APPLICATION).

EMMA is offering support for multiple network topologies, as well as all the IT Infrastructure auxiliary equipment (UPS, cooling etc) [1]. We engaged in scaling the platform, making it more flexible without forgetting the automation of the processes and included a ticketing tool which, automated or supervised by the NOC Engineers, is able to offer traceability for the incidents.

As the infrastructure evolved, our proposed platform evolved and scaled to the next level, covering the need for a Configuration Management Database [2]

[1] University POLITEHNICA of Bucharest, Romania, e-mail: mihai@roedu.net
[2] University POLITEHNICA of Bucharest, Romania, e-mail: tivanoaica@bmsit.net
[3] University POLITEHNICA of Bucharest, Romania, e-mail: arapa@bmsit.net
[4] University POLITEHNICA of Bucharest, Romania, e-mail: lucian.paiusescu@roedu.net
[5] University POLITEHNICA of Bucharest, Romania, e-mail: mihai.carabas@cs.pub.ro

and IP Address Management [3] which is the starting point of the next tools that will be integrating in our platform.

There had been different approaches, oriented only to security objects [6][7][9] where it is created a knowledge database regarding security incidents. We aim to extend these solutions in order to also store the assets of an entity and do relevant correlations in order to easily trace a security incident. With this extent we can provide better sharing of the security information [10].

This paper presents in the second section the design of the platform followed by the third section which extends the automatic discovery strategies we are using for a given IT environment. Section four is describing the feature called configuration management database and section five is presenting the security feature correlation between the current patch-level of the computers in a network and a database of vulnerabilities.

2. Design of the platform

EMMA acts as a processing and correlation module for the current data sources and security platforms. EMMA is fed with information from different vulnerability databases and assets (via SNMP). The information is stored and corelated for different activities needed in a NOC/SOC environment: monitor, alerting, recovering. EMMA is easily integrable with security products like Hive (for incident response) and MISP (for incident sharing).

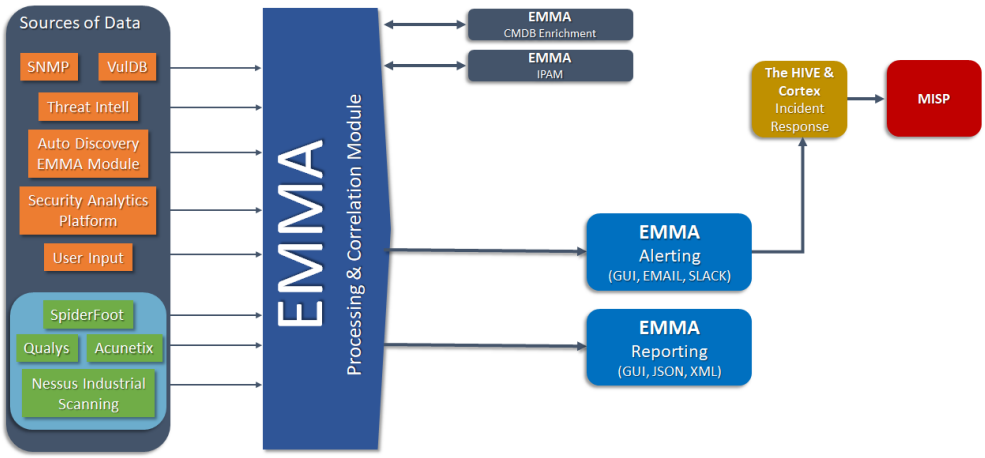

Fig. 1. Platform design

3. Discovery strategies

Nowadays, the first step for any adaptive network monitoring platform starts with the infrastructure discovery process and, given the fact that we need a

platform that can easily adapt to any challenge infrastructure design, we created EMMA as a modular application.

EMMA is able to run a discovery process that can use any information provided by a CMDB/IPAM tool that the company is already using. Also, EMMA is able to do automatic network discovery that can use various technologies based on a layered model structured on the OSI stack. In order to reduce the time and increase the performance of our software platform we are using multi-processor capabilities. The discovery process is following the layered design performing the following steps:

1. Infrastructure device and links discovery
2. Layer 2 tunnels and interface discovery
3. Additional IP addresses discovery
4. Layer 3 IP tunnels and link
5. IP routing layer protocols

In the next paragraph we will iterate through each discovery step and extend the description of the process bringing in relevant examples.

The new devices, interfaces and links found are inserted in database having a "pending approval" flag. Starting from a given host, we are using multiple algorithms and different techniques to create a map of the network mostly based on discovery protocols available (CDP, LLDP, HDP, etc). For the data that is generated in the discovery process, the platform triggers the next module that automatically finds the management and monitoring IP address of the device using a prioritizing search pattern algorithm. If the algorithm cannot retrieve the IP address, the platform will automatically use the IP address reported by the discovery protocol. The last step of the infrastructure discovery consists of adding the interfaces and devices to the database operation that leads to the physical network topology, grouping the devices with interface links as edges.

```
DEVICE_NEW:10.10.1.1
INTERFACE_NEW:10.10.1.1(DEV-NAME):2(GigabitEthernet0/0)
INTERFACE_NEW:10.10.1.1(DEV-NAME):3(GigabitEthernet0/1)
INTERFACE_NEW:10.10.1.1(DEV-NAME):4(GigabitEthernet0/2)
INTERFACE_NEW:10.10.1.1(DEV-NAME):5(Backplane-GigabitEthernet0/3)
INTERFACE_NEW:10.10.4.64(DEV-NAME):15(Tunnel101)
MNGIP_ERROR:10.0.0.6(dev-name1)
MNGIP_ERROR:10.0.0.5(dev-name2)
DEVICE_MOD:10.10.10.5(DEV-SW1)-10.10.15.18(DEV-SW1):640

runtime    0m38.308s
```

Fig. 2. Discovery procedure

After the crawl functions is finishing the discovery process, the platform is generating a report containing the following information:

- new devices, interfaces and links (added as user requested)
- not found devices, interfaces and links (no action, just reporting)
- not found IP addresses according to search patterns
- various errors required for adjusting the script to network particularities

EMMA's second step aims to discover link aggregations and L2 tunnel interfaces (LACP, ETHERCHANNEL, etc.) which are retrieved and automatically added to the database building a new layer of device interconnections. A common technology, used in many of the existing enterprise environments, is the IP Tunneling. These tunnels, in our case, are detected and added to the monitoring platform. Based on the newly added datasets, the third layer of interconnection is created.

Each routing protocol or process is discovered and together with the details and parameters are automatically added to the database for monitoring/troubleshooting, building the final layer of interconnections.

4. Configuration Management Database (CMDB)

As EMMA offers modularity, the database is structured to fit multiple scenarios allowing someone to have devices added/grouped in sites/branches, that can be further grouped by cities, in counties from different countries and, using geolocation tools we can easily place pins on maps to have a complete topology of the infrastructure (see Fig. 3).

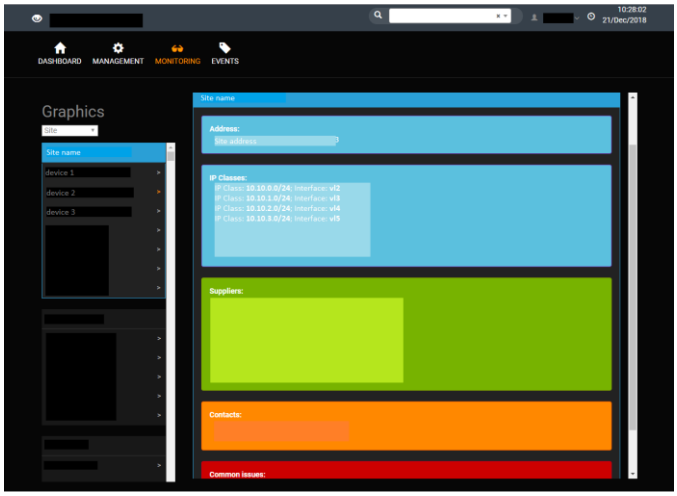

Fig. 3. The structure of the assets

For devices and sites objects created in EMMA, one can add various details, such as person in-charge, service suppliers (companies), physical and IT security related information (see Fig. 4). Also, each object has different methods of contact (email, phone). Various data can be added for contacts (ID cards, phones, emails, positions, notes) and suppliers (fiscal data, address, bank accounts, notes).

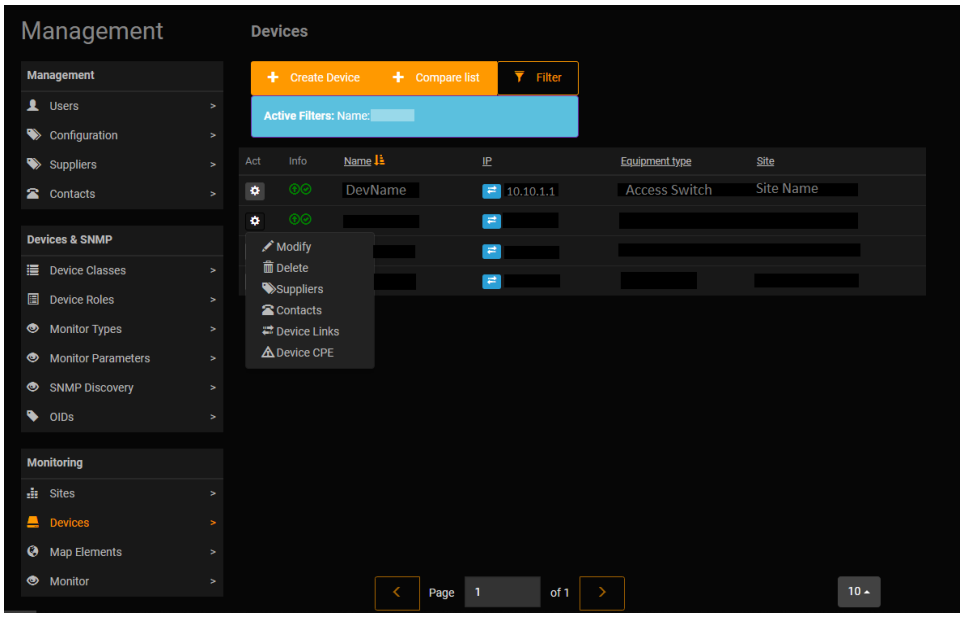

Fig. 4. Device management

For user editable data we have the following fields on the objects representing the devices: IP address, SNMP connection strings, type, site, notes, status, floor, room, rack, position, PDU port, power supplies, hardware and software serial numbers. Also, devices topology can be visual illustrated based on customer request and industry best practices. This solutions was also approached in [8]. The IP database is build based on discovered network topology taking into account the discovered VLANs (L2 and interfaces). It is based on Layer 2/3 discovered separation and it can be adjusted after customer specifications. The discovering part of the IP database is done by retrieving MAC and/or IP addresses from Layer 2/3 devices and grouping them based on customers network topology (eg. sites). The MAC - IP address pair is switch port identifiable and can be parsed based on time period.

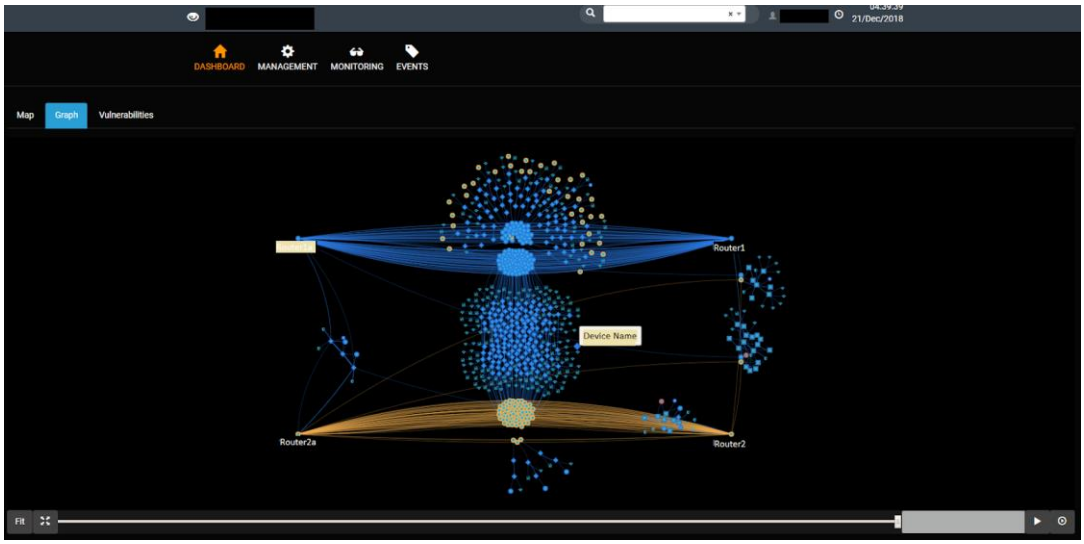

Fig. 5. Dashboard with the status of the network connections between all assets

Use-cases includes monitoring MAC - IP address changes on network ports for tracking users and easier troubleshooting tickets, determining IP addresses for DHCP-enabled devices when the MAC or used port is known [4] (see Fig. 6).

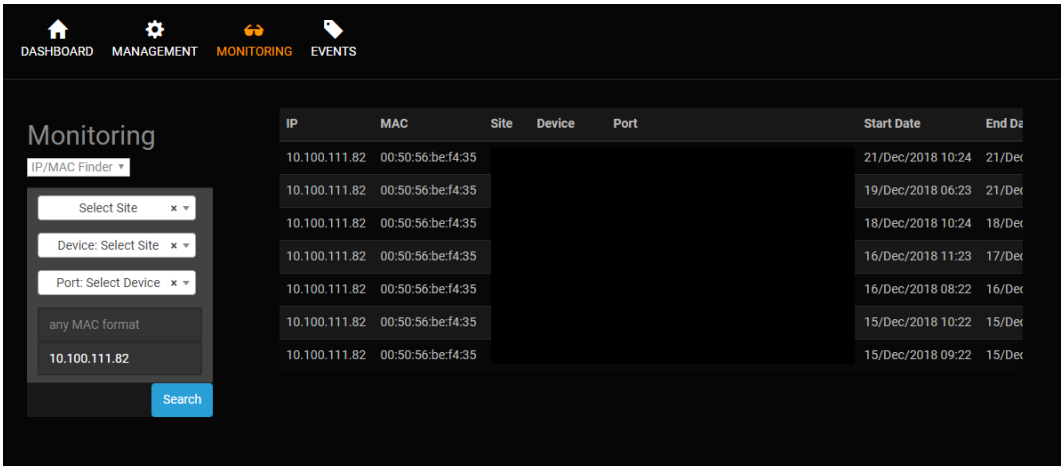

Fig. 6. IPAM – IP Address Management

5. Security vulnerabilities on the assets

We used a vulnerability database that is using CVE's with a specific scoring system in order to track down the vulnerabilities that we have in the network.

For this to be possible we developed a custom integration based on API calls to retrieve all vulnerabilities related and the match the software version found in vulnerabilities descriptions. After this operation, the databases updated with the new vulnerabilities and alerts are triggered (see Fig. 7). Because the industry practice is to address the vulnerabilities and harden the security using workarounds, the integration comes with a module for keeping track of addressed vulnerabilities for each device which contain details about the workaround (implementation time, solution, etc.).

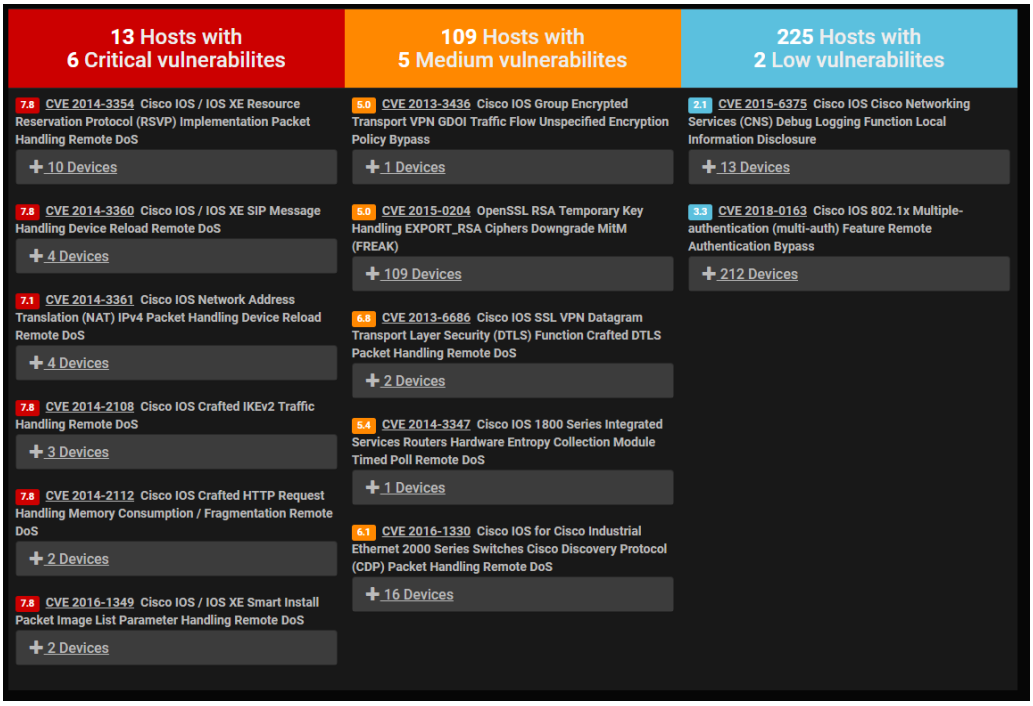

Fig. 7. Vulnerabilities on assets

Data is retrieved via SNMP using multithreading collectors for efficiency. The process is monitored for errors, overhead, performance and continuously improved. Data is stored in multiple database types – RRD, InfluxDB for time series data, SQL for specific devices data and MongoDB for free text data (vulnerabilities database) [5].

6. Conclusions and further work

This paper presented EMMA, a tool for asset discovery and management together with modules for data correlation and enrichment in order to provide monitoring and alerting capabilities to a NOC/SOC facility.

EMMA is storing asset information like IP address, software package levels, locations. We also add security information from different security feeds. In the end we are able to correlate the software package levels with the security feeds to determine which assets are vulnerable. Moreover, we store traffic flows from the network, and we correlate them with the security feeds in order to find future attacks that would take place over the network.

As future work we plan to integrate EMMA correlation engine with different machine learning algorithms to automatically detect breaches in a given network.

R E F E R E N C E S

[1]. *Madduri, Hari, et al. "A configuration management database architecture in support of IBM Service Management."* IBM Systems Journal 46.3 (2007): 441-457.

[2]. *Baron, Anthony, et al. "Configuration management database state model."* U.S. Patent No. 7,756,828. 13 Jul. 2010.

[3]. *McCloghrie, Keith, and Marshall Rose. Management Information Base for network management of TCP/IP-based internets: MIB-II.* No. RFC 1213. 1991.

[4]. *Sitaraman, Aravind, et al. "Integrated IP address management."* U.S. Patent No. 6,427,170. 30 Jul. 2002.

[5]. *Traversat, Bernard A., et al. "Transaction management in a configuration database."* U.S. Patent No. 6,115,715. 5 Sep. 2000.

[6]. *Griffin, Andrew M. "Configuration management database security."* U.S. Patent Application 13/180, 914, filed January 17, 2013.

[7]. *Xie, Michael, Robert A. May, and Jinhai Yang*. "Automated configuration of endpoint security management." U.S. Patent 9, 894,034, issued February 13, 2018.

[8]. *Rangarajan, Govindarajan, Ajoy Kumar, Narayan Kumar, Douglas Mueller, Matthew D. Connors, and Darius Wallace. "Cooperative naming for configuration items in a distributed configuration management database environment."* U.S. Patent Application 16/116,292, filed December 27, 2018.

[9]. *Shah, Mitul S., and Ruchi R. Jain. "Database security tool."* U.S. Patent Application 10/248, 805, filed April 2, 2019.

[10]. *Reybok, Richard, II Kurt Joseph Zettel, Phillip DiCorpo, Simon N. Allen, Amit Sharma, and Giora Tamir. "Network security threat intelligence sharing."* U.S. Patent Application 15/588, 152, filed November 8, 2018.