

# Securing Web Resources using RFID, Dynamic Groups, SIEM Alerting and Honeypots

Mark Deegan

*Dept. of Computing, Carlow Campus  
South East Technological University  
Rep. of Ireland*  
mark@deeganit.net

Martin Harrigan

*Dept. of Computing, Carlow Campus  
South East Technological University  
Rep. of Ireland*  
martin.harrigan@setu.ie

**Abstract—**

**Index Terms**—component, formatting, style, styling, insert

## I. INTRODUCTION

To secure a Web resource in a corporate environment, there are two modes of access to consider: an on-site user internally accessing the resource and a remote user externally accessing the same resource. In the first case, the user is physically located in a building that may be guarded by an access control system. In the second case, the user is located outside of those buildings. However, attempts to access the Web resource are often handled similarly in both cases: the physical location of the user is ignored. In this paper we describe a system that combines location with credentials when securing Web resources. Specifically, it utilises RFID tags and readers, dynamic groups in Microsoft Active Directory or a similar directory service, SIEM alerting, and honeypots. The system provides an additional layer of defence that imposes no extra burden on users.

*There is no silver bullet solution with cybersecurity;  
a layered defence is the only viable defence.*

(James Scott, ICIT, <https://www.icitech.org/>)

We deployed the system using a combination of open-source and proprietary off-the-shelf components. The components are loosely coupled and interchangeable. The approach can be adapted to a variety of corporate environments that utilise access control systems in their buildings. Furthermore, the system can be layered on top of existing infrastructure.

The system distinguishes between internal and external threats and records additional context during failed attempts to access a Web resource. Insider threats can be difficult and time-consuming to detect [1]. Our system flags occasions where a resource is accessed internally using credentials belonging to a user that is currently operating remotely. This runs counter to many existing systems where internal traffic is assumed to be safer than external traffic.

This paper is organised as follows. In Sect. II we review related work, including RFID technology, multi-factor authentication, dynamic role-based access control and the zero trust security model. In Sect. III we describe our system:, the various components and their configuration. We detail our results in Sect. IV. Specifically, we consider two use cases: an

internal user with an external threat, and an external user with an internal threat. In both cases, we demonstrate the system's ability to flag threats. Finally, we conclude in Sect. V.

## II. RELATED WORK

We categorise related work into four areas: using RFID technology to integrate physical and digital security; multi-factor authentication, specifically schemes that utilise location as one of the factors; dynamic role-based access control; and the zero trust security model or zero trust architecture.

RFID can enhance and modernise conventional approaches to access control and authentication. Clarke [2] surveys a range of transparent user authentication schemes, including schemes that rely on RFID tags and other contactless tokens. Farooq et al. [3], Larchikov et al. [4] and Woo-Garcia et al. [5] describe access control systems that employ RFID tags to differentiate between valid and invalid users. All of the systems read the RFID tags at the entrances and exits of a building. Kriplean et al. [6] deploy a building-wide RFID infrastructure with eighty RFID readers that gather fine-grained location information. They consider ways of creating utility while respecting the privacy of users. Ostojić et al. [7] deploy a similar system to manage access to a parking lot. Our system uses a Raspberry Pi connected to an RFID door entry system in a similar manner. There are many concerns surrounding RFID tags including cloning, man-in-the-middle attacks, denial-of-service attacks, communication layer weaknesses, and physical attacks (see, e.g., Ranasinghe and Cole [8]).

Ometov et al. [9] surveys multi-factor authentication (MFA) schemes: they consider various types of MFA sensors including geolocation sensors. Location-based MFA schemes, such as the one described by Ramatsakane and Leung [10], seek to balance usability and security. Suo et al. [11] use *location signatures* to secure automated vehicles. A location signature is a geo- and time-stamped message issued by a trusted device that attests to a vehicle's presence in a particular location at a given time [12]. We use the location of a user's access card as an authentication factor.

Dynamic role-based access control (DRBAC) is an extension of traditional role-based access control (RBAC) [13] that enables the automatic adjustment of user roles based on factors like context, behaviour, and risk assessment. Unlike static

RBAC, DRBAC assigns roles dynamically in response to real-time conditions, ensuring users have an appropriate access level. This enhances security and adaptability but requires real-time evaluation that can be more complex to implement. Uzun et al. [14] extend the traditional RBAC model to handle temporal and geospatial constraints. Luo et al. [15] extend the RBAC model to a cloud environment where roles are determined by the security state and network availability of the resources. Chatterjee et al. [16] describe a decentralised RBAC model that relies on a blockchain with smart contracts. They implement a proof-of-concept on the Ethereum virtual machine (EVM) and quantify its computational cost in terms of EVM gas. Finally, Liu et al. [1] survey insider threats and describe systems where host, network and contextual data can identify such threats. Our system has a related capability: it can flag insider threats by comparing the location of the target with the source of the connection.

The zero trust security model or zero trust architecture (ZTA) shifts cybersecurity defences from static, network-based perimeters to users, assets, and resources that are dynamic and perimeter-less [17]. Rose et al. [18] and Garbis and Chapman [19] define ZTA and describe its logical components. Bertino [20] highlights management and deployment as the main challenges of ZTA. Ross et al [21] show that multiple cyber resiliency techniques, can be integrated into the design and deployment of ZTA. Yao et al. [22] combine ZTA and trust-based access control (TBAC) to evaluate the trust of users and compare those evaluations against trust thresholds. In the same vein as above, Meng et al. [23] utilise a blockchain to decentralise the operation of the trusted nodes in a ZTA. Identifying the location of a user and changing their group membership based on that location, follows the principles of ZTA. By making the process fully automated and transparent, we can minimise the management and deployment overhead.

### III. METHOD

Our system comprises a variety of off-the-shelf components:

- Microsoft Active Directory with LDAP support
- Raspberry Pi connected to an RFID door entry system (see Fig. 2)
- Progress Kemp LoadMaster [24]
- Progress WhatsUp Gold [25]
- Web servers (Apache HTTP Server)

#### A. Microsoft Active Directory With LDAP Support

We used Microsoft Windows Server 2019 to run a Domain Controller (DC) (`d1.testlab.local`) for our domain (`testlab.local`). This is the primary DC for the scenario and it was setup to host Active Directory (AD) with LDAP support and the Domain Name System (DNS). To ensure that the requests for the Web resources went to the appropriate services on the Progress Kemp LoadMaster, we created corresponding DNS delegations. This is important as we wanted the external requests to be pointed to the external resources and the internal requests to be pointed to the internal resources. It also allows the LoadMaster to perform service health checks before

Active Directory Users and Computers		
File	Action	View
File	File	File
Action	Action	Action
View	View	View
Help	Help	Help
Active Directory Users and Computers		
Saved Queries		
testlab.local		
Builtin		
Computers		
Domain Controllers		
ForeignSecurityPrincipal		
Keys		
LostAndFound		
Managed Service Account		
Program Data		
System		
TestlabUsers		
Users		
NTDS Quotas		
TPM Devices		
Name	Type	Description
Administrator	User	Built-in account for ad...
Allowed RODC Password Rep...	Security Group...	Members in this group c...
Cert Publishers	Security Group...	Members of this group ...
Cloneable Domain Controll...	Security Group...	Members of this group t...
Denied RODC Password Rep...	Security Group...	Members in this group c...
DnsAdmins	Security Group...	DNS Administrators Gro...
DnsUpdateProxy	Security Group...	DNS clients who are per...
Domain Admins	Security Group...	Designated administrato...
Domain Computers	Security Group...	All workstations and ser...
Domain Controllers	Security Group...	All domain controllers i...
Domain Guests	Security Group...	All domain guests
Domain Users	Security Group...	All domain users
Enterprise Admins	Security Group...	Designated administrato...
Enterprise Key Admins	Security Group...	Members of this group ...
Enterprise Read-only Domai...	Security Group...	Members of this group ...
External	Security Group...	
Group Policy Creator Owners	Security Group...	Members in this group c...
Guest	User	Built-in account for gue...
Internal	Security Group...	
Jane Doe	User	
John Doe	User	
Key Admins	Security Group...	Members of this group ...
krbtgt	User	Key Distribution Center ...
Protected Users	Security Group...	Members of this group ...
RAS and IAS Servers	Security Group...	Servers in this group can...
Read-only Domain Controll...	Security Group...	Members of this group ...
Schema Admins	Security Group...	Designated administrato...

Fig. 1. We created two users, Jane Doe and John Doe, and two groups, Internal and External, in Active Directory.

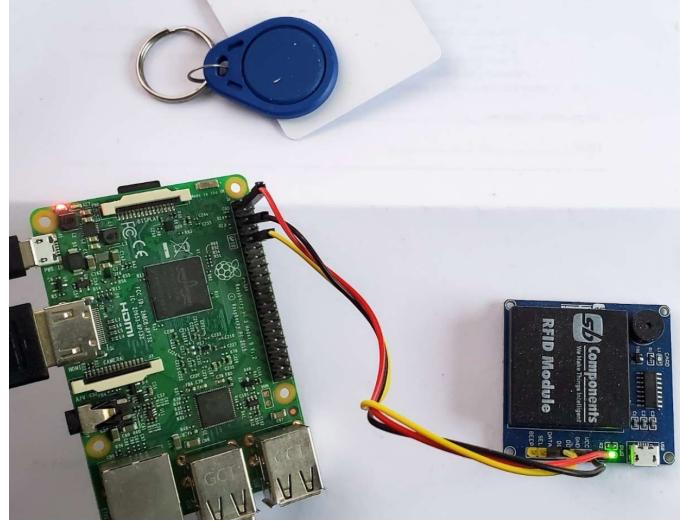


Fig. 2. The Raspberry Pi was connected to an RFID door entry system. We created a Python script that read the identification details from swiped cards and interacted with Active Directory.

responding to DNS requests thus preventing IP addresses being returned when resources are unavailable. We added two users, Jane Doe and John Doe, and two groups, Internal and External, to AD (see Fig. 1).

#### B. Raspberry Pi and RFID Door Entry System

The Raspberry Pi is connected to an RFID door entry system. When a user performs a card swipe, a Python script running on the Raspberry Pi reads the identification details from the card and remotely changes the group membership of

the user in AD. If the user is entering the building their group membership is changed from External to Internal; if they are leaving the building it is changed from Internal to External. It logs the event to a SIEM service (see Sect. III-E).

### C. Progress Kemp LoadMaster

Progress Kemp LoadMaster (LM) is a reverse proxy and load balancer. It has many capabilities but for this scenario we are interested in the Edge Security Pack (ESP), the source IP blacklist from the GEO component, and the Web Application Firewall (WAF). We use the ESP for Single Sign-On (SSO) for HTTP(S) services and to communicate with the AD for both for logon and group memberships. This pre-authenticates a user before they gain access to a resource. We also enabled *group steering* on the ESP: this allows LM to send traffic to particular services based on their group membership in AD and goes beyond the normal use of groups to simply allow or deny access.

We created two steering groups associated with the Internal and External groups in AD. We created Perl Compatible Regular Expression (PCRE) rules to match the authorisation cookies and steer the requests to the appropriate services (see Fig. 3).

The login page that is displayed by the ESP is the same for both *valid* and *invalid* access attempts. A valid access attempt occurs when a user's group and request are both internal or both external; otherwise the access attempt is invalid. In both cases a log is sent to a SIEM service (see Sect. III-E). This helps with threat hunting as a threat actor will get the same login page for the honeypot as with the valid site. The honeypot can gather the details of the access attempt without being discovered.

The GEO component performs DNS resolution and service health checks before returning a result. We created GEO DNS entries for `internal.testlab.local` and `external.testlab.local` (see Fig. 4). We also used an IP blacklist that is updated daily to withhold DNS results from anyone on the list.

Additionally, we enabled the Intrusion Detection System (IDS) and Intrusion Prevention Systems (IPS) on LM. This includes rules defined by the SNORT community [26] and enables the SNORT rule filtering on the Layer 7 HTTP engine to check for any known bad requests. Figure 5 shows the configuration highlighted in green. There is also an option to prevent access via whitelists and blacklists highlighted in blue.

Finally, we enabled the Web Application Firewall (WAF) and the Open Web Application Security Project (OWASP) core rule set. This rule set performs anomaly scoring and identifies, for each request, the probability that it is malicious. The core rule set protects against SQL injection, cross-site scripting, remote code execution, buffer overflows, known vulnerabilities, and many other vectors of attack. We configured the WAF with source IP reputation blocking enabled which uses a global IP reputation list that is updated daily. Using the MaxMind [27] and the GEO component, it identifies the country of the source

request and it can be configured to block specific countries or regions.

### D. Web Servers

Our Web servers are hosted on virtual running Debian and a default installation of the Apache HTTP Server. The landing page is our “valid access” page and represents our secured Web resource. The “invalid access” page is served by a Flask application. It records the username and source IP of all requests and sends those details to a SIEM service (see next section).

### E. Progress WhatsUp Gold

The Raspberry Pi, LM, and Flask application send logs to a SIEM service. We use Progress WhatsUp Gold (WUG) (see Fig. 6). In normal operation the events from the Raspberry Pi, RFID door entry system, and LM are logged. In cases where a user's credentials may be compromised, the Flask application logs an event with high priority.

## IV. RESULTS

We configured the system as described in the previous section. The goal of the study was to show the feasibility of the integration between the various components, and to demonstrate that dynamic group membership based on real-world location can secure Web resources. We performed two tests (Sect. IV-A and Sect. IV-B) to demonstrate this aspect of the system.

### A. Internal User with External Threat

In the first case the user, John Doe, enters his office during normal working hours and swipes his access card at the door using an RFID tag. His group membership is set to Internal and the user can then access the resource internally. Meanwhile an external threat actor attempts to login from outside the office while the user is at work. They are denied access and they have their IP address logged to the SIEM service (WUG) as a breach attempt. This requires no extra overhead on the user to secure his credentials. The timeline of events is as follows:

- 1) John Doe enters his office and swipes his access card.
- 2) The user's group membership is changed from External to Internal. This event is logged to the SIEM service from the Raspberry Pi.
- 3) When the user gets to his desk they access the Web resource internally.
- 4) The DNS points to the LM for DNS resolution of the Web resource and since it is an internal request the user is sent to the internal service.
- 5) The ESP requires the user to login.
- 6) The user's group membership is checked by the ESP SSO system and they are connected to the appropriate Web resource.
- 7) An external threat actor attempts to access the Web resource using John Doe's credentials.
- 8) The DNS points the threat actor to the LM for DNS resolution.

The screenshot shows the LoadMaster Content Rules interface. On the left, there's a navigation sidebar with options like Home, Virtual Services, Global Balancing, Manage FQDNs, Statistics, and Help. The main content area is titled "Content Matching Rules". It lists two entries:

Name	Type	Options	Header	Pattern	In Use	Operation
Group1_steering	RegEx	Ignore Case	Cookie	X-Kemp-STEERING=1	✓ 2	<a href="#">Modify</a> <a href="#">Delete</a> <a href="#">Duplicate</a>
Group2_Steering	RegEx	Ignore Case	Cookie	X-Kemp-STEERING=2	✓ 2	<a href="#">Modify</a> <a href="#">Delete</a> <a href="#">Duplicate</a>

Fig. 3. ...

The screenshot shows the LoadMaster Global Fully Qualified Names interface. The left sidebar includes sections for Home, Virtual Services, Global Balancing, Manage FQDNs (which is selected), Statistics, and Help. The main area is titled "Configured Fully Qualified Domain Names". It displays two entries:

Fully Qualified Domain Name	Type	IP Address	Cluster	Checker	Availability	Requests/s	Parameters	Operation
external.testlab.local	Round Robin	192.168.32.203		ICMP Ping	✓ Up	0		<a href="#">Modify</a> <a href="#">Delete</a>
internal.testlab.local	Round Robin	192.168.32.205		ICMP Ping	✓ Up	0		<a href="#">Modify</a> <a href="#">Delete</a>

Fig. 4. We configured two FQDNs for `internal.testlab.local` and `external.testlab.local` using LoadMaster's GEO component.

The screenshot shows the LoadMaster Advanced Properties interface. The left sidebar has sections for Home, Virtual Services, Global Balancing, Manage FQDNs, Statistics, and Help. The main area contains several configuration sections:

- Content Switching:** Enabled (highlighted in green). Options include Show Selection Rules, Show Header Rules, and Show Body Modification Rules.
- HTTP Selection Rules:** A dropdown menu with options like "Intrusion Handling", "Drop Connection", and "Warnings".
- HTTP Header Modifications:** A section with "Add Header to Request" and "Copy Header in Request" fields, along with "Set Header" and "Set Headers" buttons.
- Response Body Modification:** A dropdown menu with "Add HTTP Headers" set to "X-Forwarded-For (No Via)".
- Enable HTTP/2 Stack:** An unchecked checkbox.
- Enable Caching:** An unchecked checkbox.
- Enable Compression:** An unchecked checkbox.
- Detect Malicious Requests:** A dropdown menu with "Intrusion Handling" checked, "Drop Connection" selected, and "Warnings" checked.
- Not Available Redirection Handling:** Fields for "Error Code" and "Redirect URL", with "Set Redirect URL" and "Set Server Address" buttons.
- Add a Port 80 Redirector VS:** A field for "Redirection URL" containing "`https://%h%`".
- Default Gateway:** A field with "Set Default Gateway" button.
- Service Specific Access Control:** A dropdown menu with "Access Control" selected.

Fig. 5. We enabled IDS/IPS in LoadMaster as an additional layer of defence for little configuration.

Date	Source	Syslog Type	Payload
04/17/2023 8:53:13 pm	RFID	Any Syslog	Message=<14>RFID read: None
04/17/2023 8:53:12 pm	RFID	Any Syslog	Message=<14>RFID read: Added user JohnDoe to Internal group
04/17/2023 8:53:11 pm	RFID	Any Syslog	Message=<14>RFID read: 020047BE758E
04/17/2023 8:53:11 pm	RFID	Any Syslog	Message=<14>RFID read: JohnDoe
04/17/2023 8:53:10 pm	RFID	Any Syslog	Message=<14>RFID read:

Fig. 6. ...

Date	Source	Syslog Type	Payload
04/19/2023 3:59:17 pm	192.168.32.236	Unsolicited	Message=<14>Critical - Invalid Access Attempt by username john doe@testlab.local from clientip 192.168.32.236

Fig. 7. ...

- 9) Using the correct credentials, the external threat actor logs in via the ESP SSO page.
- 10) The group membership is read as Internal, but the source is external, so the threat actor is directed to a “server unavailable” page and their IP address is logged to the SIEM service (see Fig. 7).

#### B. External User with Internal Threat

In the second case we are concerned with internal threats: rather than credentials being leaked externally, they are accessed by a threat internally, e.g., someone who may have physical access to the user’s desk. The user, Jane Doe, leaves the office for lunch and swipes her access card on the RFID reader on the way out. Her group membership is set to External. Meanwhile an internal threat actor attempts to login from inside the office while the user is away. They are denied access and an event is logged to the SIEM service (WUG) as a breach attempt. The timeline of events is as follows:

- 1) Jane Doe leaves her office and swipes his access card.
- 2) The user’s group membership is changed from Internal to External. This event is logged to the SIEM service from the Raspberry Pi.
- 3) An internal threat actor attempts to access the Web resource using Jane Doe’s credentials.
- 4) The DNS points the threat actor to the LM for DNS resolution.
- 5) Using the correct credentials, the internal threat actor logs in via the ESP SSO page.

- 6) The group membership is read as External, but the source is internal, so the threat actor is directed to a “server unavailable” page. The event is logged to the SIEM service.

## V. CONCLUSION

### REFERENCES

- [1] L. Liu, O. De Vel, Q.-L. Han, J. Zhang, and Y. Xiang, “Detecting and preventing cyber insider threats: A survey,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 1397–1417, 2018.
- [2] N. Clarke, *Transparent User Authentication: Biometrics, RFID and Behavioural Profiling*, 1st ed. Springer, 2011.
- [3] U. Farooq, M. ul Hasan, M. Amar, A. Hanif, and M. U. Asad, “RFID based security and access control system,” *IACSIT International Journal of Engineering and Technology*, vol. 6, no. 4, pp. 309–314, 2014.
- [4] A. Larchikov, S. Panasenko, A. V. Pimenov, and P. Timofeev, “Combining RFID-based physical access control systems with digital signature systems to increase their security,” in *International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2014, pp. 100–103.
- [5] R. M. Woo-Garcia, and U. H. Lomeli-Dorantes, F. López-Huerta, A. L. Herrera-May, and J. Martínez-Castillo, “Design and implementation of a system access control by RFID,” in *IEEE International Engineering Summit (IE-Summit)*, 2016, pp. 1–4.
- [6] T. Kriplean, E. Welbourne, N. Khoussainova, V. Rastogi, M. Balazinska, G. Borriello, T. Kohno, and D. Suciu, “Physical access control for captured RFID data,” *IEEE Pervasive Computing*, vol. 6, no. 4, pp. 48–55, 2007.
- [7] G. Ostojić, S. Stankovski, and M. V. Jovanovic, “Implementation of RFID technology in parking lot access control system,” in *Annual RFID Eurasia*, 2007, pp. 1–5.
- [8] D. C. Ranasinghe and P. H. Cole, “Confronting security and privacy threats in modern RFID,” in *The Asilomar Conference on Signals, Systems and Computers*, 2006, pp. 2058–2064.

- [9] A. Ometov, S. Bezzateev, N. Mäkitalo, S. Andreev, T. Mikkonen, and Y. Koucheryavy, “Multi-factor authentication: A survey,” *Cryptography*, vol. 2, no. 1, 2018.
- [10] K. I. Ramatsakane and W. S. Leung, “Pick location security: Seamless integrated multi-factor authentication,” in *IST-Africa Week Conference (IST-Africa)*, 2017, pp. 1–10.
- [11] D. Suo, J. Moore, M. Boesch, K. Post, and S. E. Sarma, “Location-based schemes for mitigating cyber threats on connected and automated vehicles: A survey and design framework,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 4, pp. 2919–2937, 2022.
- [12] C. Chen, X. Wang, W. Han, and B. Zang, “A robust detection of the sybil attack in urban VANETs,” in *IEEE International Conference on Distributed Computing Systems*, 2009, pp. 270–276.
- [13] V. Franqueira and R. Wieringa, “Role-based access control in retrospect,” *Computer (IEEE Computer Society)*, vol. 45, no. 6, pp. 81–88, 2012.
- [14] E. Uzun, V. Atluri, S. Sural, J. Vaidya, G. Parlato, A. L. Ferrara, and M. Parthasarathy, “Analyzing temporal role based access control models,” in *ACM Symposium on Access Control Models and Technologies*, 2012, pp. 177–186.
- [15] J. Luo, H. Wang, X. Gong, and T. Li, “A novel role-based access control model in cloud environments,” *International Journal of Computational Intelligence Systems*, vol. 9, no. 1, pp. 1–9, 2016.
- [16] A. Chatterjee, Y. Pitroda, and M. Parmar, “Dynamic role-based access control for decentralized applications,” in *International Conference on Blockchain*. Springer, 2020, pp. 185–197.
- [17] N. F. Syed, S. W. Shah, A. Shaghaghi, A. Anwar, Z. Baig, and R. Doss, “Zero trust architecture (ZTA): A comprehensive survey,” *IEEE Access*, vol. 10, pp. 57143–57179, 2022.
- [18] S. Rose, O. Borchart, S. Mitchell, and S. Connelly, “Zero trust architecture,” [https://tsapps.nist.gov/publication/get\\_pdf.cfm?pub\\_id=930420](https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=930420), 2020.
- [19] J. Garbis and J. W. Chapman, *Zero Trust Security*, 1st ed. Apress, 2021.
- [20] E. Bertino, “Zero trust architecture: Does it help?” *IEEE Security & Privacy*, vol. 19, no. 5, pp. 95–96, 2021.
- [21] R. Ross, V. Pillitteri, R. Graubart, D. Bodeau, and R. McQuaid, “Developing cyber-resilient systems: A systems security engineering approach,” <https://csrc.nist.gov/pubs/sp/800/160/v2/r1/final>, 2021.
- [22] Q. Yao, Q. Wang, X. Zhang, and J. Fei, “Dynamic access control and authorization system based on zero-trust architecture,” in *International Conference on Control, Robotics and Intelligent System*. ACM, 2021, pp. 123–127.
- [23] L. Meng, D. Huang, J. An, and X. Zhou, “A continuous authentication protocol without trust authority for zero trust architecture,” *China Communications*, vol. 19, no. 8, pp. 198–213, 2022.
- [24] Progress Kemp, “LoadMaster,” <https://kemptechnologies.com/>.
- [25] ———, “WhatsUp Gold,” <https://kemptechnologies.com/>.
- [26] Cisco, “Snort,” <https://www.snort.org/>.
- [27] MaxMind, “GeoIP2 Databases,” <https://www.maxmind.com/en/solutions/ip-geolocation-databases-api-services>.