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

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 [1] surveys a range of transparent user authentication schemes, including schemes that rely on RFID tags and other contactless tokens. Farooq el al. [2], Larchikov et al. [3] and Woo-Garcia et al. [4] 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. [5] 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. [6] 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 [7]).

Ometov et al. [8] 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 [9], seek to balance usability and security. Suo et al. [10] 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 [11]. 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) [12] that enables the automatic adjustment of user roles based on factors like context, behavior, 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. [13] extend the traditional RBAC model to handle temporal and geospatial constraints. Luo et al. [14] 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. [15] 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. [16] 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.

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 [17]
- Progress WhatsUp Gold [18]
- 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 responding to DNS requests thus preventing IP addresses being returned when resources are unavailable. We added two users,

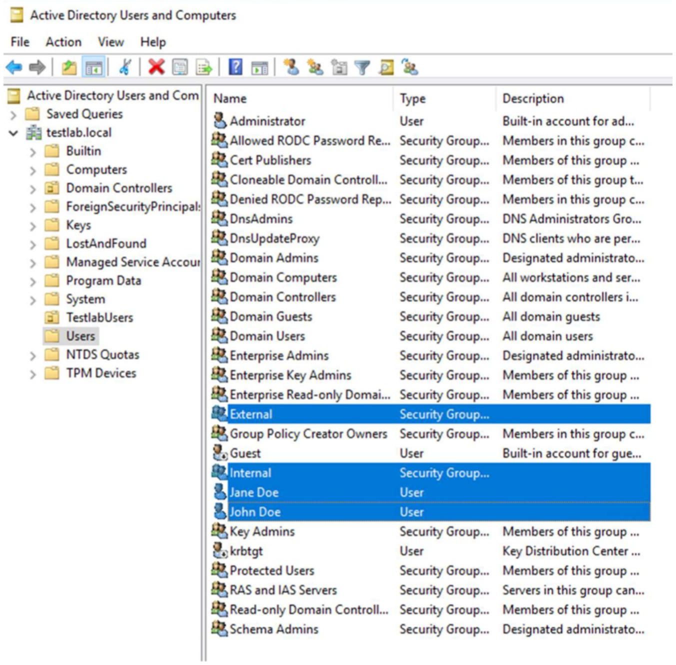

Fig. 1. We created two users, Jane Doe and John Doe, and two groups, Internal and External, in Active Directory.

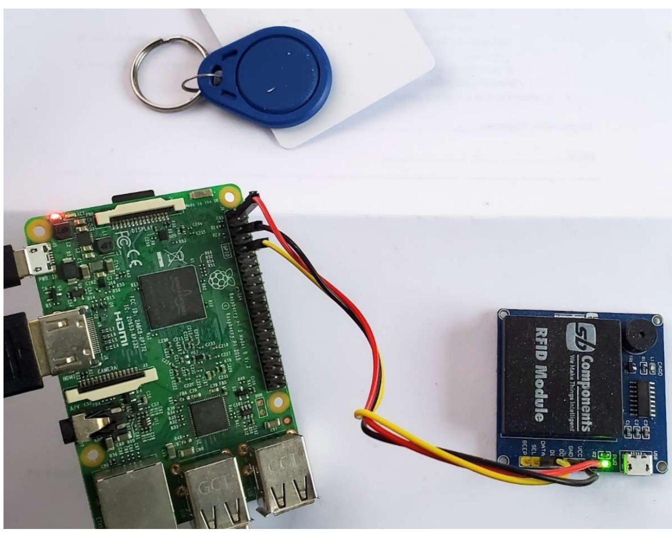

Fig. 2. The Raspberry Pi was connected to an RFID door entry system. We created a Python script that read the identification details from swiped cards and interacted with Active Directory.

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 honey pot as with the valid site. The honey pot 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 [19] 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 [20] and the GEO component, it identifies the country of the source request and it can be configured to block specific countries or regions.

Fig. 3. ...

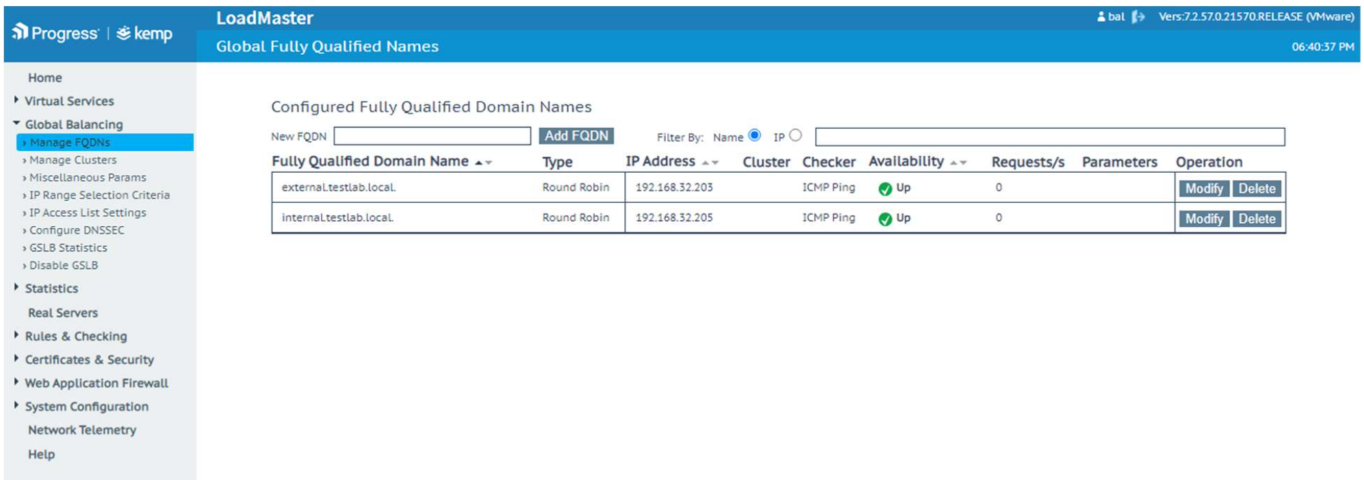

Fig. 4. We configured two FQDNs for `internal.testlab.local` and `external.testlab.local` using LoadMaster's GEO component.

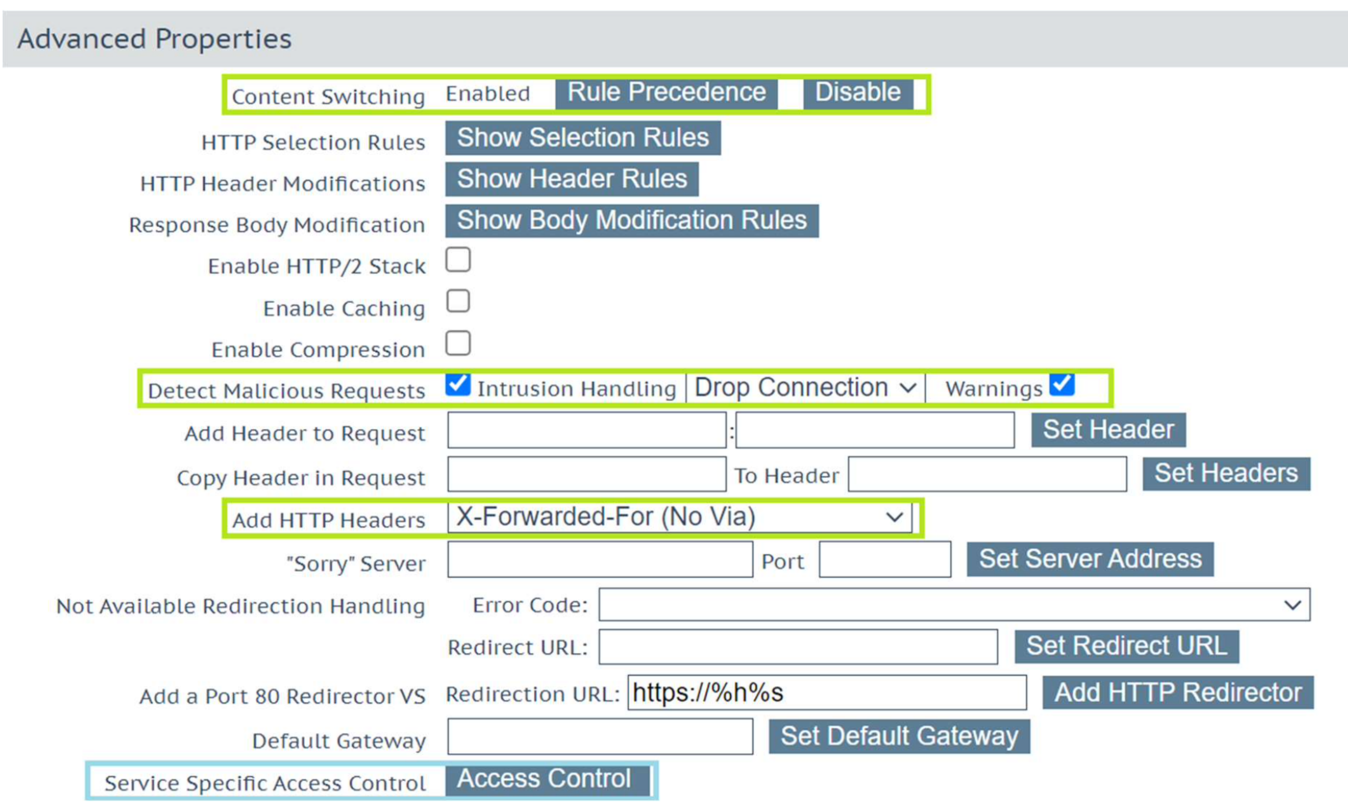

Fig. 5. We enabled IDS/IPS in LoadMaster as an additional layer of defence for little configuration.

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.
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] N. Clarke, *Transparent User Authentication: Biometrics, RFID and Behavioural Profiling*, 1st ed. Springer, 2011.

[2] 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.

[3] 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.

[4] R. M. Woo-Garcia, anbd 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.

[5] 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.

[6] 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.

[7] 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.

[8] A. Ometov, S. Bezzateev, N. Mäkitalo, S. Andreev, T. Mikkonen, and Y. Koucheryavy, "Multi-factor authentication: A survey," *Cryptography*, vol. 2, no. 1, 2018.

[9] 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.

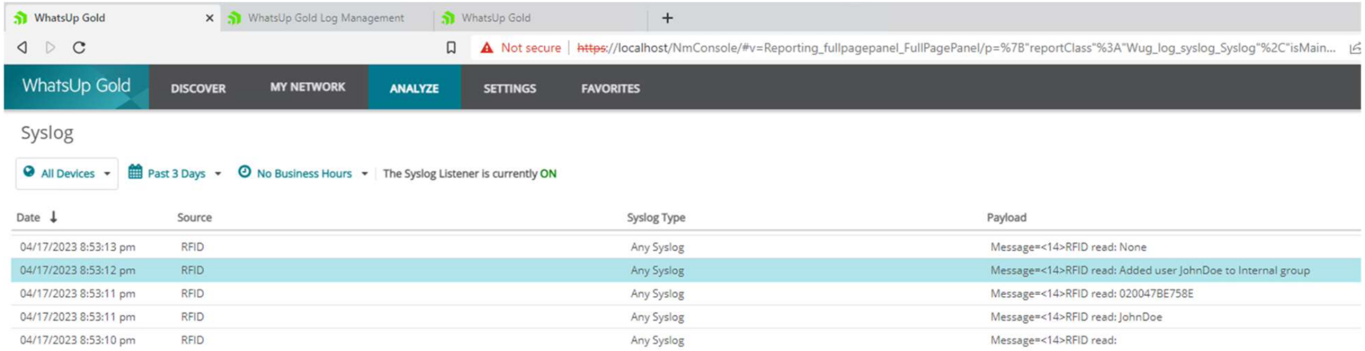

Fig. 6. ...

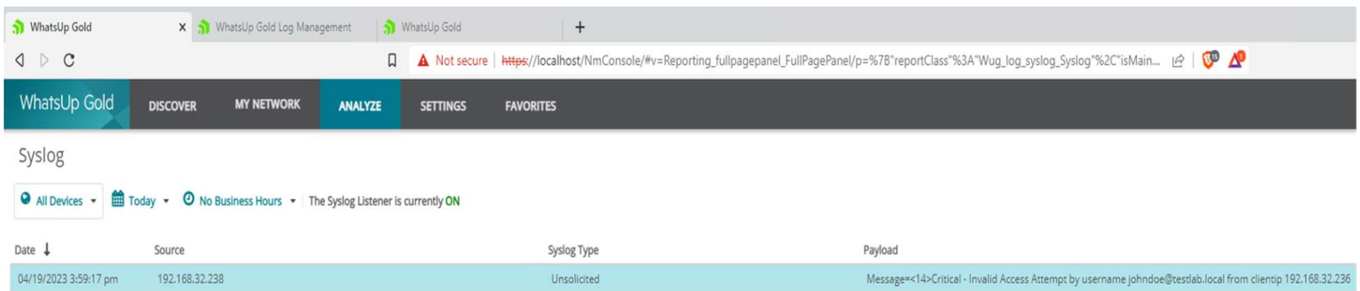

Fig. 7. ...

[10] 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.

[11] 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.

[12] V. Franqueira and R. Wieringa, “Role-based access control in retrospect,” *Computer (IEEE Computer Society)*, vol. 45, no. 6, pp. 81–88, 2012.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] Progress Kemp, “LoadMaster,” https://kemptechnologies.com/.

[18] ——, “WhatsUp Gold,” https://kemptechnologies.com/.

[19] Cisco, “Snort,” https://www.snort.org/.

[20] MaxMind, “GeoIP2 Databases,” https://www.maxmind.com/en/solutions/ip-geolocation-databases-api-services.