

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

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*

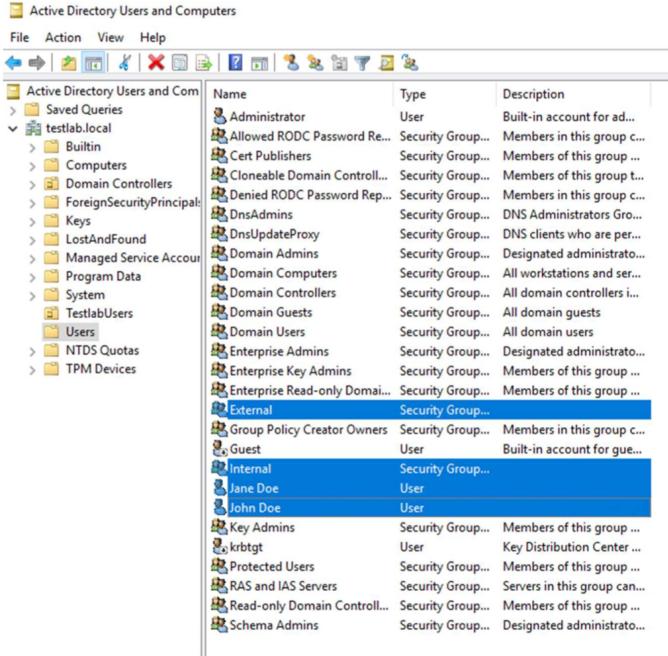


Fig. 1. We created two users, Jane Doe and John Doe, and two groups, Internal and External, in Active Directory.

Abstract—

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

I. INTRODUCTION

II. RELATED WORK

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 [1]
- Progress WhatsUp Gold [2]
- Web servers (Apache HTTP Server)

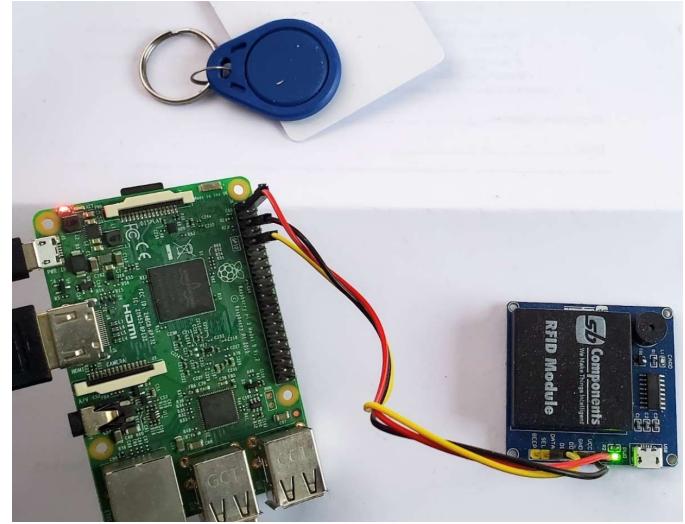


Fig. 2. The Raspberry Pi was connected to an RFID door entry system. We created a Python script that read the identification details from swiped cards and interacted with Active Directory.

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, 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 [?] 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 [?] 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 their office during normal working hours and swipes their card at the door using an RFID tag. Their group membership is set to Internal and 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 username and IP address logged to the SIEM service (WUG) as a breach attempt. This requires no extra overhead on the user to secure their credentials. The timeline of events is as follows:

- 1) John Doe enters their office and swipes their 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 their 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.

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	Modify Delete Duplicate
Group2_Steering	RegEx	Ignore Case	Cookie	X-Kemp-STEERING=2	✓ 2	Modify Delete Duplicate

Fig. 3. ...

The screenshot shows the LoadMaster Global Fully Qualified Names interface. The left sidebar includes options for 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		Modify Delete
internal.testlab.local	Round Robin	192.168.32.205		ICMP Ping	✓ Up	0		Modify Delete

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.
- HTTP Header Modifications:** A dropdown menu.
- Response Body Modification:** A dropdown menu.
- Enable HTTP/2 Stack:** An unchecked checkbox.
- Enable Caching:** An unchecked checkbox.
- Enable Compression:** An unchecked checkbox.
- Detect Malicious Requests:** A section with checkboxes for Intrusion Handling (checked), Drop Connection (selected), and Warnings (checked).
- Add Header to Request:** Fields for header name and value, and a Set Header button.
- Copy Header in Request:** Fields for source header and target header, and a Set Headers button.
- Add HTTP Headers:** A dropdown menu currently set to "X-Forwarded-For (No Via)" (highlighted in green).
- "Sorry" Server:** Fields for server name and port, and a Set Server Address button.
- Not Available Redirection Handling:** Fields for Error Code and Redirect URL, and a Set Redirect URL button.
- Add a Port 80 Redirector VS:** Fields for Redirection URL (set to `https://%h%`) and an Add HTTP Redirector button.
- Default Gateway:** A field with a Set Default Gateway button.
- Service Specific Access Control:** A dropdown menu currently set to "Access Control" (highlighted in green).

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	192.168.32.238	Unsolicited	Message=<14>Critical - Invalid Access Attempt by username johndoe@testlab.local from clientip 192.168.32.236

Fig. 7. ...

- 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 username and source IP address are logged to the SIEM service (see Fig. 7).

B. External User with Internal Threat

V. CONCLUSION

REFERENCES

- [1] Progress Kemp, “LoadMaster,” <https://kemptechnologies.com/>.
- [2] ——, “WhatsUp Gold,” <https://kemptechnologies.com/>.