2011 International Conference on Information and Network Technology
IPCSIT vol.4 (2011) © (2011) IACSIT Press, Singapore

Identity Engines: Ensuring Authentication, Authorization, and Audit across Network

Sumit Kalra [1], Vikas Grover [1] and Vipin Kumar Goyal [2]

1 System Engineer, Infosys Technologies Ltd.

2 Student, UIET, Panjab University

Abstract. For over a decade now, RADIUS servers have been a mainstay of dial-up and VPN access control. But in the RADIUS server, Authentication of the clients is shallow i.e. it doesn't provide a proper way to implement Authentication at large scale. In this paper, we show that inclusion of Identity Engine with RADIUS server provide a way to implement the Authentication in secure manner.

1.Introduction

For over a decade now, RADIUS servers have been a mainstay of dial-up and VPN access control. The rather inconspicuous RADIUS server, perhaps better known as that beige, general-purpose PC collecting dust in the corner of your data center, has proved sufficient for performing basic duties like validating passwords and granting network access. But while these servers have been diligently chugging away at their tasks, the world of networking and security technology has evolved substantially, leaving the current generation of RADIUS servers in the dust.

The emergence of wired and wireless 802.1X network authentication, combined with NAC, has outstripped the capabilities of the current-generation RADIUS servers. Fortunately, Identity Engines has built the future of RADIUS servers in its next-generation, hardened RADIUS appliance, the Ignition Server. Following are the reasons to implement Identity Engine with RADIUS server:

- Missing 'Authentication' from AAA(Authorization, Authentication and Accounting).
- Heterogeneous RADIUS server in heterogeneous Network.
- Need of multiple directories to provide security.
- Need of multiple RADIUS server.
- RADIUS server is critical element of Network.
- Consistent Group naming, multiple directories with same groups.
- RADIUS server is security sensitive.
- Authenticating employees against LDAP is fine, but what about your guests?
- It's not user or device; it's user and device.
- RADIUS is just the beginning.

2. Missing 'Authentication' from AAA(Authorization, Authentication and Accounting)

Accounting, authentication, and authorization are the cornerstones of a RADIUS server's functionality. When clients connect to a network, authentication validates who they are; authorization dictates what resources clients can use, and accounting tracks what clients have done. Frustratingly, for most networks today the middle "A", authorization, is missing; feasible network authorization remains more dream than reality.

Today's unauthenticated networks and legacy RADIUS servers are incapable of performing such functions. In order to provide authorization, a RADIUS server needs to have a more in-depth conversation with the network-edge device through which the user connects, and this conversation must be based on a far more in-depth policy.

Today's RADIUS servers simply can't implement this policy. Identity Engines' Ignition Server was built to consider such a policy routine.

At the heart of Ignition Server is a flexible policy engine based on the XACML standard. It allows you to write rules any way you want leveraging all the information available from network devices and all the work your applications group has put into your back-end user directories. Simple Boolean expressions can be linked together to form very complex policies such as: Students enrolled in Calculus 201 should be disallowed access to the network when attempting access from Branham Hall on November 12th between the hours of two and four PM. Access at all other times and all other locations should be accepted with authorization rights of the "student" role.

3. Heterogeneous RADIUS Server in Heterogeneous Network.

Chances are your remote access VPN gateways, firewalls, Ethernet switches, and WLAN APs are not from the same vendor. Even if they are, each type of device speaks RADIUS in a different way; they ask different questions and expect different responses.

RADIUS allows these different conversations through a flexible mechanism known as RADIUS attributes and RADIUS vendor-specific attributes (VSAs). Most RADIUS servers support some limited attribute and VSA functionality, but, since these RADIUS servers focus primarily on authentication, this functionality is neither easy to use nor robust. For example, one RADIUS server's VSA handling requires importing VSA definitions from an external database manually.

Ignition Server was built to perform authorization for all types of network gear and includes robust attribute and VSA handling. A built-in vendor library provides attribute definitions for all the popular vendors and equipment. New vendors, products, and VSAs can be added easily, and you can create templates that aggregate several commonly used attributes into a single object for use throughout the system, for both inbound and outbound RADIUS communication.

4. Author Need of Multiple Directories to Provide Security.

For many organizations, the promise of a central user directory is quite compelling: having a single source from which to retrieve all the attributes and groups of your user community promises manageability, consistency, and auditability. In reality though, whether through mergers and acquisitions, business division politics, or the natural evolution of most IT departments, many mid-size and nearly all larger organizations will end up supporting more than one directory. Multiple directories create a unique set of challenges.

Ignition Server acts as an arbitration point between, on the one hand, all your network devices that require users to authenticate and, on the other hand, all your user directories that ultimately verify users' credentials. One can even set up Ignition Server to use portions of more than one directory in the same authentication request. For example, one might want to authenticate critical users with hardware token. Using an ordinary RADIUS server, all the information about the user would need to be stored in the same place the token is validated. By contrast, Ignition Server can validate the token password against the token server but still use server's LDAP directory for the entire group and attribute information about the user. This avoids costly data synchronization between the two directories or worse sacrificing RADIUS functionality because the data requested is not available.

5. Need of multiple RADIUS Server

There are a number of other limitations that force organizations to deploy more RADIUS servers than they ought to. A lack of RADIUS-server flexibility is the most common cause. This happens when the RADIUS server lacks the flexibility to serve multiple types of switch gear simultaneously, forcing the IT team to install many RADIUS servers, each in a dedicated, single-purpose deployment.

Unlike traditional RADIUS servers, the Ignition Server is designed to accommodate disparate types of switchgear in a single installation. Ignition Server supports over 200 virtual RADIUS servers within a single, highly-available appliance form factor. Each virtual instance supports its own credential types, policy rules, and identity routing. With Ignition Server, you can achieve the original goal of a RADIUS deployment: having a single, authoritative authentication and authorization broker, while also gaining the deployment flexibility that today's heterogeneous networks demand.

5.1 RADIUS Server is Critical Element of Network

There are plenty of services on your network that are essential. The minute one of these services fails, your network functionality is compromised. In an authenticated network, RADIUS is one of these essential services.

If your RADIUS server is down, incoming users can't log on, and users with current sessions can't re-authenticate. So why is your RADIUS server treated like a second-class citizen in terms of reliability and availability? Ignition Server provides an appliance form factor built from the ground up for high availability. To ensure your RADIUS deployment remains running around the clock, simply link two Ignition Servers using the dedicated HA interface, and your network access policies and RADIUS configuration are immediately synchronized.

5.2 Consistent Group Naming, Multiple Directories with Same Groups

One of the challenges in dealing with multiple user directories in an organization is inconsistency in group naming. This is problematic for RADIUS deployments because it means "Engineering- Group" isn't always "Engineering-Group". Consider a policy that says:

All engineers in the entire organization have access to the test lab wireless network.

The policy looks pretty straightforward. But what happens if someone has multiple directories? One directory groups its engineers in the "Development Group" while another groups its engineers in the "Engineering-Group". The person has to write two rules to enforce that simple policy. Add another directory, and now three rules are required. Then, when person finally get around to consolidating the groups and directories, he has to edit his rules to use the new group-naming scheme.

With Ignition Server, all the groups and attributes in his policy are mapped to a common naming scheme controlled by the Ignition Server administrators. This feature, called a virtual group, lets the person create one "Engineering" group where he can use to represent dozens of groups across dozens of directories.

5.3 RADIUS Server is Security Sensitive.

By far the most common method of deploying a RADIUS server is to install it on a general purpose PC. This is the same approach that was common in the mid-1990s for deploying firewalls. Today nearly everyone deploys appliance-based firewalls, instead, simplifying ongoing maintenance and security for the device. The same problems that afflicted PC-based firewall deployments now afflict PC-based RADIUS deployments. Deploying RADIUS on an insecure operating system means one need to secure the operating system and maintain it while also maintaining the RADIUS server software.

Ignition Server is built from the ground up as an appliance. It uses a stripped-down and hardened BSD kernel with an encrypted file system to keep sensitive user data and policies safe. The tamper evident physical hardware alerts server to possible intrusions. There's only one set of firmware to maintain and the system is locked-down by default. Logs can be viewed locally or exported in CSV format or via syslog for importing into the security information management (SIM) tool of apt choice.

5.4 Authenticating Employees against LDAP is fine, but What about Your Guests?

RADIUS was built to check passwords for dial-in users, but today there is a huge variety in types of users. The type that is growing most significantly is temporary users. Temporary users come in many forms: vendors, contractors, trainees, customers, partners, and more. A common thread among temporary users is their need for some form of network access, but only for a specified period of time.

Identity Engines' Ignition Guest Manager (IGM) relieves this bottleneck. Built as an extensible J2EE web application, IGM communicates with the Ignition Server and provides a simple web interface built for the task of maintaining guest accounts. Guests may be restricted to specific zones of the network and to specific time periods, and they may be required to connect via specific access methods. All of this is accomplished with a delegated administration model that allows staff such as front desk receptionists to create accounts. IGM also provides a full audit trail of who created the user, and when and where the user connected.

5.5 It's not User or Device; it's User and Device.

Increasingly, organizations are tasked with authenticating devices as well as users. Existing RADIUS offerings can handle this, but without much grace. Many allow servers to authenticate a machine based on its MAC address as a way to bypass user authentication, but just checking that an address is on a list represents a huge sacrifice in functionality compared with the precise rules one can write in a user policy. The most common requirement that cannot be met using existing RADIUS servers is an assigned-device policy. This sort of policy allows the user to connect only if he or she has successfully authenticated and is using a company-provided laptop or other device.

Using the Ignition Server, it's easy to set up device access policies and assigned-device policies. The access rules one write can evaluate attributes of the device and/or the user. This is a huge leap beyond what's offered by legacy systems that can only ask, "Is the device registered?" In contrast, the Ignition Server lets the person write policies that state which assets can connect to which parts of his network and, optionally, which users may connect using which devices.

Ignition Server also makes it easy to merge the policies of users and devices to ensure that only specific types of users can use specific assets at specific times of the day and from prescribed network locations. Ignition Server accommodates such business policies, reducing the risk of unauthorized individuals accessing the network.

5.6 RADIUS is just the Beginning.

When RADIUS was in its nascent stages, a typical deployment included a RADIUS server with a built-in database and a network access server that offered the modem connections. Those two entities were the entire solution. Today's networks are more functional and secure. They are also more complex. Which require more powerful and simple RADIUS server.

Identity Engines lets the person deploy 802.1X on today's complex networks. The integrated Ignition product suite simplifies deployment of a next generation, identity-enabled network:

• **Ignition Guest Manager™** lets the person authenticate and audit guest network sessions with the same level of security he applies to his other user accounts, and it lets front-desk staff safely manage these accounts.

• **Ignition AutoConnect™** lets his end-users quickly configure their laptops and other devices to connect to the 802.1X-secured network, while minimizing calls to his support desk.

• **Ignition Portal™** gives his non-802.1X capable devices a way to authenticate and connect using the same policies, 802.1X-enabled community uses.

• **Ignition Posture Module™** is a NAP- and TNC-compatible agent that checks the health of connecting devices. It integrates tightly with the industry's most widely supported, open source 802.1X supplicant, the Open1X Xsupplicant.

• **Ignition Reports™** gives him historical, audit-grade reporting of every network access event.

6. Conclusion

Existing RADIUS server has a number of limitations which create complexity when it comes to implement Authentication at large scale with heterogeneous elements in Network. But the next generation of Radius server i.e. Ignition server of Identity Engine makes it easier to implement. Using the Ignition server, it's possible to accommodate different level of users with flexible naming convention including the guest

users and keep the information about each and every activity with their timing and point of access in Network. It combines the Users Access policies with the Device Access policies resulting into more secure system.

7. Acknowledgements

Please We are grateful to all persons who were directly or indirectly involve in our summer training in UIET in May-June 2009.Our special thanks to Mr. Amandeep, Mr. Satish and Mr. Manmohan whoguided us at each point of our research. With their proper support, we were inspired to do this research project with interest.We are grateful to our lecturer Mr. Naveen Aggarwal, who introduced this kind of opportunity to us and selected for this project.

8. References

[1] Alan T. Dekok. Rdeploying RADIUS : Practice and Principles for AAA solutions Co-founder and leader of FreeRADIUS Servr Project

[2] Jonathan Hassell RADIUS

[3] Identity Engine. M10 Reasons your RADIUS Server Needs a Refresh