LogRhythm®
The Security Intelligence Company

DISSECTING A CLOUD ATTACK

Securing Azure with AzLog

A Randy Franklin Smith white paper commissioned by LogRhythm

Table of Contents

Dissecting a Cloud Attack: Securing Azure with Azlog

As organizations go through a digital transformation, many are moving to the cloud to remain competitive. The need to be constantly accessible and available to customers, partners, and employees alike, makes the use of cloud services an obvious choice. Unfortunately, those advantages come with unique security challenges that expose an organization to additional risk.

The risk the cloud creates is real. Approximately 93 percent of organizations utilize cloud services of some kind, and 74 percent of them store some or all of their sensitive information in the cloud. With unauthorized access to sensitive data being the number one concern of organizations today[1], it's critical to have visibility into the management, configuration of, and access to your organizations cloud environment.

Microsoft's Azure cloud infrastructure services are quickly growing, gaining six times as much market share from Q4 2016 to Q4 2017 as its competitors (including Amazon Web Services[2]). The performance-focused, scalable, and highly available cloud platform has been a valuable choice for those organizations looking to extend their on-premises data center into the cloud. Additionally, Azure maintains robust integration points through the syncing of Azure Active Directory (AAD) with on-premises Active Directory (AD), the use of AD Federation Services, and even Office 365.

Security Logging and the Azure Cloud

Just because data, applications, and systems exist in Azure doesn't mean your commitment to security, ability to monitor, and need to achieve compliance are any less. First, you need to collect security events, configuration changes, and access logs. Then you must centralize this data within a security information and event management (SIEM) platform to achieve visibility and maintain security.

To help you better understand what's necessary and possible regarding logging and visibility of your Azure environment, we'll take a look at an example of how a typical attack takes place, how to identify progression through the Cyber Attack Lifecycle using Azure's AzLog functionality, and how to detect attackers moving laterally between the cloud and your on-premise network.

Turning Data into Actionable Insights: LogRhythm

Every time IT extends the scope of its operational environment, the security team needs to expand the scope of its monitoring immediately and automatically. When extending to the cloud, depending on who manages the infrastructure, OS, and applications being utilized, the difficulty of maintaining visibility potentially magnifies.

LogRhythm works to establish and maintain visibility by centralizing and normalizing diverse data from critical sources across an environment. It then uses advanced correlation and security analytics to turn data into actionable insight for the security team.

Look for insights from LogRhythm throughout this paper to learn how to better secure your cloud environment.

[1] McAfee, Building Trust in a Cloudy Sky Report (2017)

[2] Cloud Growth Rate Increases; Amazon, Microsoft & Google all Gain Market Share, Synergy Research Group, Feb 2, 2018. https://www.srgresearch.com/articles/cloud-growth-rate-increases-amazon-microsoft-google-all-gain-market-share

Dissecting a Cloud Attack: Cyber Attack Lifecycle

Attacks today, while in a constant state of evolution, tend to follow a textbook set of actions. Monitoring for these actions can assist IT organizations in providing better protection. If you can understand this process, known as the Cyber Attack Lifecycle or Cyber Kill Chain, the better you can detect and respond to such malicious activity.

Whether the intended goal is to steal data, to infect with ransomware, to disrupt business through data corruption, the steps to accomplish the end goal largely look the same. In the cloud, many of these steps are even easier to take, given the external accessibility to Azure and other clouds. Let's dive into each.

Figure 1: Cyber Attack Lifecycle

Initial Compromise

An attacker needs to establish a foothold within your organization. For an on-prem attack, it's best represented by a compromised endpoint. But in the cloud, it may be represented by a single compromised account. In either case, the foothold can be attained through a number of malicious actions. Phishing attacks are still alive and well today, with malware present in every 1 of 131 emails[3]. Additionally, simple social engineering scams targeting users are common and effective. Recently, Office 365 users have been tricked into divulging their passwords through a spoofed sign-in prompt.

But, initial compromise doesn't stop there. After all, the goal isn't to just gain access to a single account; in the end, the actor has a specific threat action in mind. So, threat actors seek to obtain and compromise privileged credentials up to, and including, the role of administrator. Tools like Mimikatz are used to expose credential artifacts in memory, such as password hashes (which can be used in a pass-the-hash attack), Kerberos tickets (which can be subject to cracking), or clear text passwords. Additionally, threat actors use native tools to "live off the land" in an effort to not draw attention to themselves.

LogRhythm Insights: Detect PowerShell Encoding

PowerShell is a powerful ally for hackers. It can be used to both download and invoke Mimikatz. As a result, many organizations choose to put execution policies in place to limit the use of PowerShell. Hackers can get around this policy through base64-encoded PowerShell parameters.

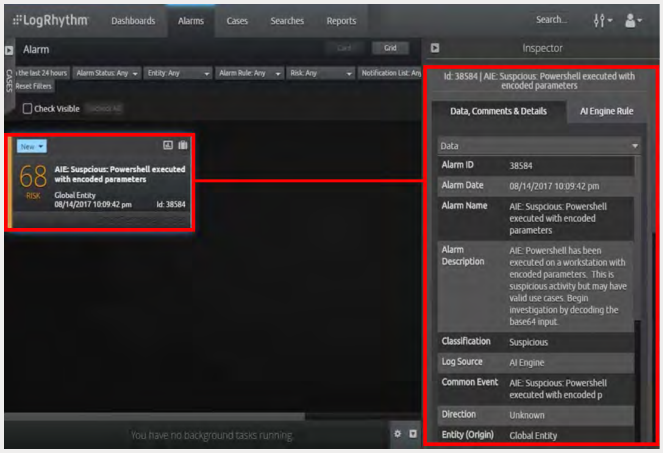

Figure 2: LogRhythm Can Alert When PowerShell Runs with Encoded Parameters

Normal use of PowerShell does not require the use of encoding. Encoding is typically only used when the script author doesn't want to divulge parameters (such as credentials) as part of the command string. To catch malicious activity, an LogRhythm AI Engine rule can be created to identify and alert the security team when PowerShell runs with encoded parameters.

Additionally, an automated SmartResponse™ can be created to instantly glean more information about the attack. It can further decode the encoded string providing more detail on the nature of the PowerShell script.

Eighty-one percent of data breaches involve the misuse of credentials because they are necessary to achieve an attacker's goal. Without them, the threat actor is stuck on a single endpoint with access to limited information. Once a set of privileged credentials[4] are obtained, and both stealth and persistence have been established the next step is to expand the threat surface within the organization by connecting to additional endpoints.

[3] Symantec, Internet Security Report (2017)
[4] Verizon, Data Breach Investigations Report (2017)

Lateral Movement

Gaining access to additional systems is a step towards finding the right database or virtual machine (VM) that provides attackers the data they seek. Even when ransomware is the method of attack, new variants seek to use mapped drives and Server Message Block (SMB) connections to encrypt data local to the infected endpoint, as well as on as many additional systems as possible.

From the initial compromise, the attacker can use these credentials to access Azure, AAD, Office 365, and on-prem resources, including AD. If the compromised credentials are privileged enough, additional credentials can be created to establish persistence and continued stealth access to Azure.

If a foothold has been established on an endpoint, threat actors can connect to additional endpoints using several methods, including SMB, Remote Desktop Protocol (RDP), Windows Management Instrumentation (WMI), and PowerShell remoting. It's also critical to note that you should have access to security log data both on-premises and in Azure. Because an initial compromise in either environment can potentially facilitate lateral movement to the other, it's necessary to have both sets of security data to quickly identify an initial compromise.

LogRhythm Insights: Identifying Privileged Account Creation

Movement within your organization requires two things: connectivity to additional endpoints and credentials with which to connect. Threat actors understand only having a single account is risky, as once abnormal activity is detected, it's usually disabled. So, threat actors seek ways to create additional accounts within AD or AAD as a means of establishing persistence within your network. If one of their accounts is discovered, they have many more lying in wait.

LogRhythm can easily identify the creation of accounts both in AD and AAD. In this example, the user, *Bruce*, was compromised using his credentials found in memory. A quick search for all activity for the user shows the creation of an account in AAD.

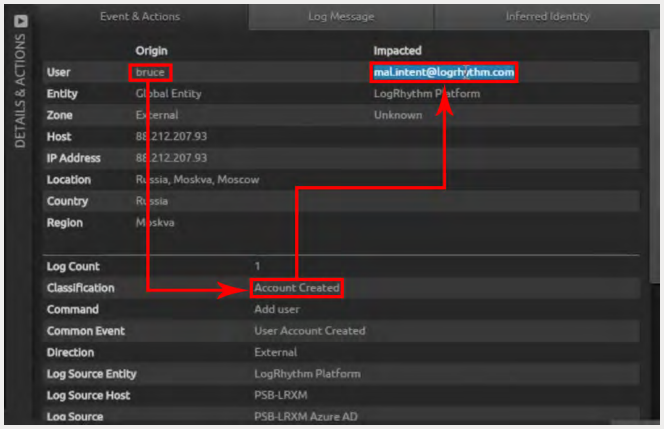

Figure 3: LogRhythm Can Spot When Backdoor Accounts are Being Created in AAD

The user then created an account, *mal.intent@logrhythm.com*, for use as a backdoor should their access to the bruce account be terminated. Further examination can expose actions, such as adding this user to a privileged group that grants access to AAD, VMs, DCs, or servers.

LogRhythm AI Engine rules can detect lateral movement across the entire environment. In this example, any combination of behavior classified as suspicious that correlates with the creation of a new account would trigger a high-risk alarm. The security operations center (SOC) could then quickly detect and respond to the incident before target attainment and threat action can occur.

Lateral movement continues in a cyclical pattern repeating itself along with initial compromise tactics, seeking out additional credentials in memory and connecting to as many endpoints and resources as possible. This continues until a target has been attained.

Target Attainment

Once the threat actor is inside the environment with elevated privilege and moving laterally, the next goal is to identify and access data targets of value. In Azure, with the right credentials, data stores can be enumerated – as can VMs, databases, and so forth – in an attempt to identify accounts that have access to those resources. Attaining a target is not just about connecting to a given system. It can be a complex set of actions performed together to locate data of value.

[4] Verizon, Data Breach Investigations Report (2017)

If you're still unclear as to what kinds of actions a threat actor would use to attain a target, imagine yourself logged onto a server within foreign network, with the goal of locating and exfiltrating customer records. Next, think about all the steps you'd take to determine whether there is a VM, a database, a set of tools, or another system that can be taken advantage of to help you achieve the goal.

LogRhythm Insights: Viewing User Activity

There is no set script of actions a threat actor takes to reach their target, so it's important to have an ability to audit what actions they've taken. These actions can be anything from connecting to a server, modifying settings, performing searches, and application-specific actions. The goal for IT is to have full visibility into user activity across your environment.

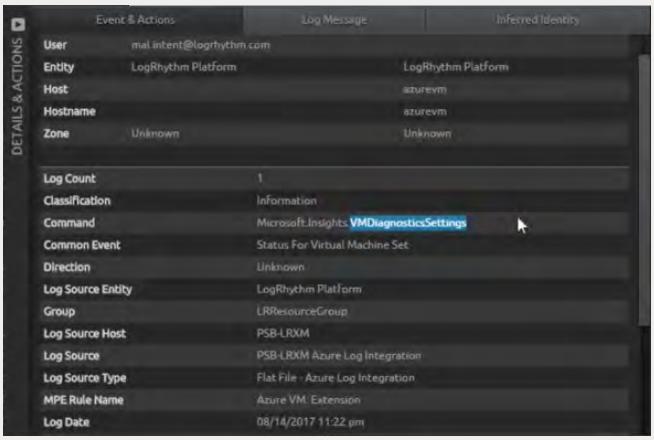

Figure 4: LogRhythm Intelligently Consolidates and Normalizes Disparate Activity Data

LogRhythm normalizes the wide variety of log data into consistent metadata fields, enabling powerful search capabilities across multiple sources. This provides comprehensive visibility into user behavior. In this example, a search on the user *mal.intent* reveals all activity related to the account. Activities such as auditing a database, changing passwords to a VM, and modifying the configuration of a VM are all easily found to provide context around the attack specifics.

LogRhythm AI Engine rules detect common patterns of target attainment, such as abnormal process activity, authentication failures, modification of system time, and clearing log files. AzLog's VM Event Log capabilities enable LogRhythm to audit a dynamic Azure PaaS environment just as it would a conventional set of Windows hosts.

Once a data set has been identified by a threat actor, the next step is for them to act.

Threat Action

The actions taken at this step differ, depending on the intent of the threat actor and the targeted organization. In the case where ransomware is the intended threat action, the identified target could be all Office documents, PDFs, and so on. In the case of data exfiltration, the target could be credit card data, personnel data (including health or SSN data), customer records, or credentials. For example, with data corruption, the intended target could be specific data sets, auditing capabilities, or even the endpoints themselves (as in the case of Saudi Aramco where 35,000 machines were wiped out in a suspected case of espionage).

LogRhythm Insights: Leveraging the Environment

In this final step of the Cyber Attack Lifecycle, detection is crucial. LogRhythm AI Engine progression rules tie together previously detected, but disparate, security events that share a common indicator. Alone, these events aren't necessarily indicative of an attack, but through advanced correlation, LogRhythm tells a story that enables a SOC to quickly and efficiently react.

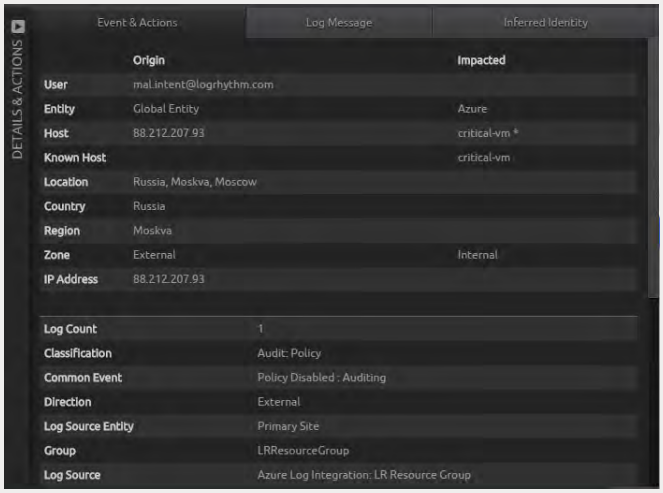

Figure 5: LogRhythm Detects All Types of Threat Actions, Such as the Disabling of Auditing

LogRhythm detects threat action across three categories:

Exfiltration: where threat actors might access Azure data stores or key vaults

Disruption: which might involve modifying or creating new Azure VMs for computationally expensive activity, such as botnets or bitcoin mining

Corruption:

- Covert: where threat actors remove audit trails, replace downloads to include malware, or host phishing sites that leverage a domain's reputation to avoid proxy filters
- Overt: where threat actions could install ransomware, deface a website, or remove critical files

LogRhythm uses both Control Plane and Data Plane Azure logs to identify these actions and spot occurrences when the environment itself is being repurposed for advance malicious activity.

But what if, like most organizations, an increasing amount of your information and IT resources are stored in Azure? When the attacker's footsteps lead to the cloud, the trail doesn't need to grow cold. LogRhythm's support for Azure's AzLog allows you to stay on the trail no matter where it takes you.

Cyber Attack Lifecycle: Conclusion

The reality is, threat actors will use your entire infrastructure to discover and exploit weaknesses. Cloud platforms are no exception. While an attack timeline can vary from minutes to months, defeating threats depends on the ability to effectively detect and track a threat actor through the Cyber Attack Lifecycle. Achieving this comprehensive visibility requires that all security log data, configuration changes, and access logs from the cloud are gathered then centralized within a SIEM for correlation.

Logging with Azure and AzLog

Until recently, access to security log data in Azure was either not available, or it was only partially accessible via purpose-specific portals or difficult-to-implement APIs. Any security professional will tell you that both options are less than optimal, particularly when attempting to maintain security in real time.

Azure maintains a fairly compete set of activity logging, including:

Control Plane

- Administrative changes performed in Azure Resource Manager, PowerShell, and Azure Command Line Interface
- Administrative changes in AAD

Data Plane

- Changes to Azure data and data security through Azure Diagnostics
- Changes to Azure VMs, including the logs found within guest operating systems
- Network, resource, and VM behavior analysis using Azure Security Center

Having this data available for correlation and analysis in your SIEM makes it even more valuable. Luckily, you can now easily ingest Azure logs into your SIEM using *Azure Log Integration*, known as AzLog.

Introduction to AzLog

Microsoft realized the need to provide its customers with visibility into their Azure instances, so it created AzLog. AzLog provides and ability to integrate the raw logs from Azure into an on-premises SIEM solution. This integration requires a system (physical or virtual) to run the Azure Log Integration Service and a set of Azure resources that need to be monitored.

The consolidated logged event types mentioned earlier can be outputted to a few formats, depending on the particular log source (more on this later), including:

- Syslog server
- Common event format (CEF)
- Log event extended format (LEEF)
- JavaScript Object Notation (JSON) files
- Forwarded events into the Event Log on the Windows system running AzLog

The format you choose will depend on how you intend on centralizing, consolidating, analyzing, and addressing the log data.

Let's look at some of the specific data AzLog makes accessible.

Azure AD Audit Logs

All tasks related to the management of user and group accounts, as well as all other objects stored in Azure AD are logged in Azure AD Audit Logs. This is the same data available in the Office 365 unified audit log if you have Office 365 implemented for the instance of AAD in question. Tactically speaking, the data you'll likely be pulling from this log is account management and authentication events.

Output Formats Available: Log data can be sent to either an *AzureActiveDirectoryJSON* folder on the server running AzLog, to a Syslog server, or saved as a Syslog file.

LogRhythm Insights: User and Entity Behavioral Analytics (UEBA)

Azure AD logs provide critical insight into user behavior. Everything, from where and when a user signs in to which systems a user accesses, is available via AD logs. This activity can be fed into LogRhythm CloudAI, which uses machine learning to detect anomalous user activity and assist investigation.

As Azure is often exposed to the internet, sign-in activity plays a critical role to determine if a user may have been compromised. In this example, LogRhythm CloudAI can analyze new host and location information from Azure activity logs to identify attempted brute-force authentication, credential theft, or compromised API access.

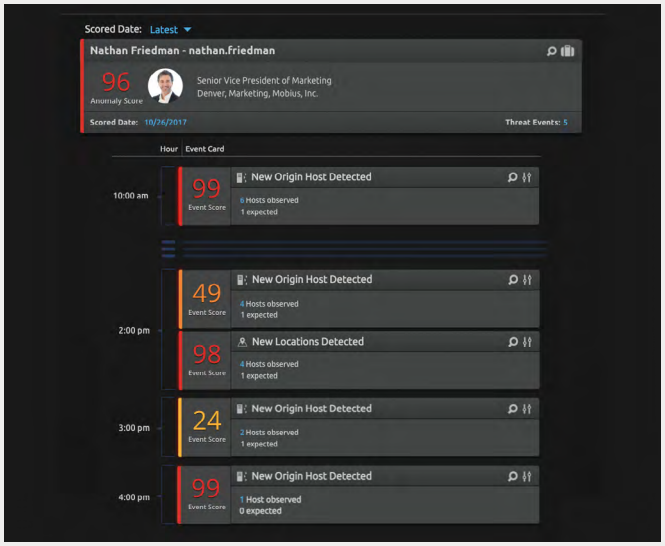

Figure 6: LogRhythm CloudAI User Timeline

Azure Resource Manager Logs

All administrative activity (including automated activity) performed within Azure is stored Resource Manager Logs. Example administrative activity includes spinning up or shutting down VMs, creating storage accounts, and modifying permissions. Whether actions are taken from within the ARM portal, using PowerShell, or using Azure CLI, any kind of changes made to the Azure environment itself are logged.

Output Formats Available: Log data can be sent to either an *AzureResourceManagerJSON* folder on the server running AzLog, to a Syslog server, or saved as a Syslog file.

Azure Diagnostics

The name of this service needs a bit of clarification. While there are some bits of performance and diagnostic detail provided, what's more important is its focus on logging what is happening on the data plane of Azure. Azure Diagnostics is where Microsoft logs access to resources, so it becomes the data source of information to determine who creates, deletes, modifies, writes, or downloads data to a storage account.

Output Formats Available: Log data can be sent to either an *AzureDiagnosticsJson* folder on the server running AzLog, to a Syslog server, or saved as a Syslog file.

Azure Security Center Alerts

An agent is installed within VM and sends the events to an Azure storage account. By integrating AzLog with the Azure Diagnostics service, AzLog (when configured with the storage account name and access key) uses Windows Event Forwarding to push events into the *Forwarded Events* Log on the computer running AzLog.

Output Formats Available: Log data is only available as a forwarded event.

Azure-Based VMs

Azure does do some of its own security monitoring and makes this data available in the Azure Security Center. AzLog can pull the following kinds of data:

- Virtual Machine Behavioral Analysis (VMBA): Events such as process creation and logons are logged.
- Network Analysis: Detail around outgoing traffic, or communication with a malicious machine that could be suspect would be logged here.
- Contextual Information: Other actions that provide context, such as clearing a log, plugging in an unknown PNP device, and other alerts that are not actionable.

Output Formats Available: Log data can be sent to either an *AzureSecurityCenterJson* folder on the server running AzLog, to a Syslog server, or saved as a Syslog file.

LogRhythm Insights: AzLog Installation

The installation procedure for AzLog is well-documented by Microsoft. So, rather than walk you through the steps, below are the basic steps and links to documentation to get you started.

1. Install the AzLog service on your on-prem system that can forward TCP syslog to your SIEM.
2. Connect AzLog to Azure.
3. Authorize AzLog access to your Azure subscription.
4. Configure any Syslog servers as log destinations.
5. Enable Azure Diagnostics on any VMs and configure log forwarding.
6. Configure your SIEM to access and collect all AzLog data (if not already accomplished when defining your syslog servers).

You can find the detailed steps and additional information below:

- Azure Log Integration with Azure Diagnostics Logging and Windows Event Forwarding
- How to Get Your Security Center Alerts in Azure Log Integration
- Azure Log Integration FAQ
- LogRhythm Cloud Monitoring

Maintaining a Unified Central View of Security Events

As your organization begins to use Azure as part of your extended data center environment, your cloud infrastructure and applications are just as vulnerable as you on-premises solutions. You can protect them by achieving comprehensive visibility through a SIEM.

While Microsoft has taken steps to provide a vast wealth of security log data within Azure, the key to achieving full security visibility is by connecting your SIEM to Azure through AzLog. By using logs in JSON format, you can achieve complete visibility and contextual insight into suspicious activity from a unified point of view. This will yield actionable intelligence to make your environment—on-prem or in the cloud—more secure.

To learn more about expanding visibility and securing your cloud environment, schedule a live online demonstration with a LogRhythm expert: https://logrhythm.com/schedule-online-demo/

About LogRhythm

LogRhythm is a world leader in NextGen SIEM, empowering organizations on six continents to successfully reduce risk by rapidly detecting, responding to and neutralizing damaging cyberthreats. The LogRhythm platform combines user and entity behavior analytics (UEBA), network traffic and behavior analytics (NTBA) and security automation & orchestration (SAO) in a single end-to-end solution. LogRhythm's Threat Lifecycle Management (TLM) workflow serves as the foundation for the AI-enabled security operations center (SOC), helping customers measurably secure their cloud, physical and virtual infrastructures for both IT and OT environments. Built for security professionals by security professionals, the LogRhythm platform has won many accolades, including being positioned as a Leader in Gartner's SIEM Magic Quadrant.

www.logrhythm.com

About Randy Franklin Smith

Randy Franklin Smith is an internationally recognized expert on the security and control of Windows and AD security. Randy publishes www.UltimateWindowsSecurity.com and wrote The Windows Server 2008 Security Log Revealed—the only book devoted to the Windows security log. Randy is the creator of LOGbinder software, which makes cryptic application logs understandable and available to log-management and SIEM solutions. As a Certified Information Systems Auditor, Randy performs security reviews for clients ranging from small, privately held firms to Fortune 500 companies, national, and international organizations. Randy is also a Microsoft Security Most Valuable Professional.

Contact us:

TOLL FREE 1-866-384-0713
FAX (303) 413-8791
EMAIL info@logrhythm.com
Worldwide HQ, 4780 Pearl East Circle, Boulder CO, 80301

LogRhythm®
The Security Intelligence Company