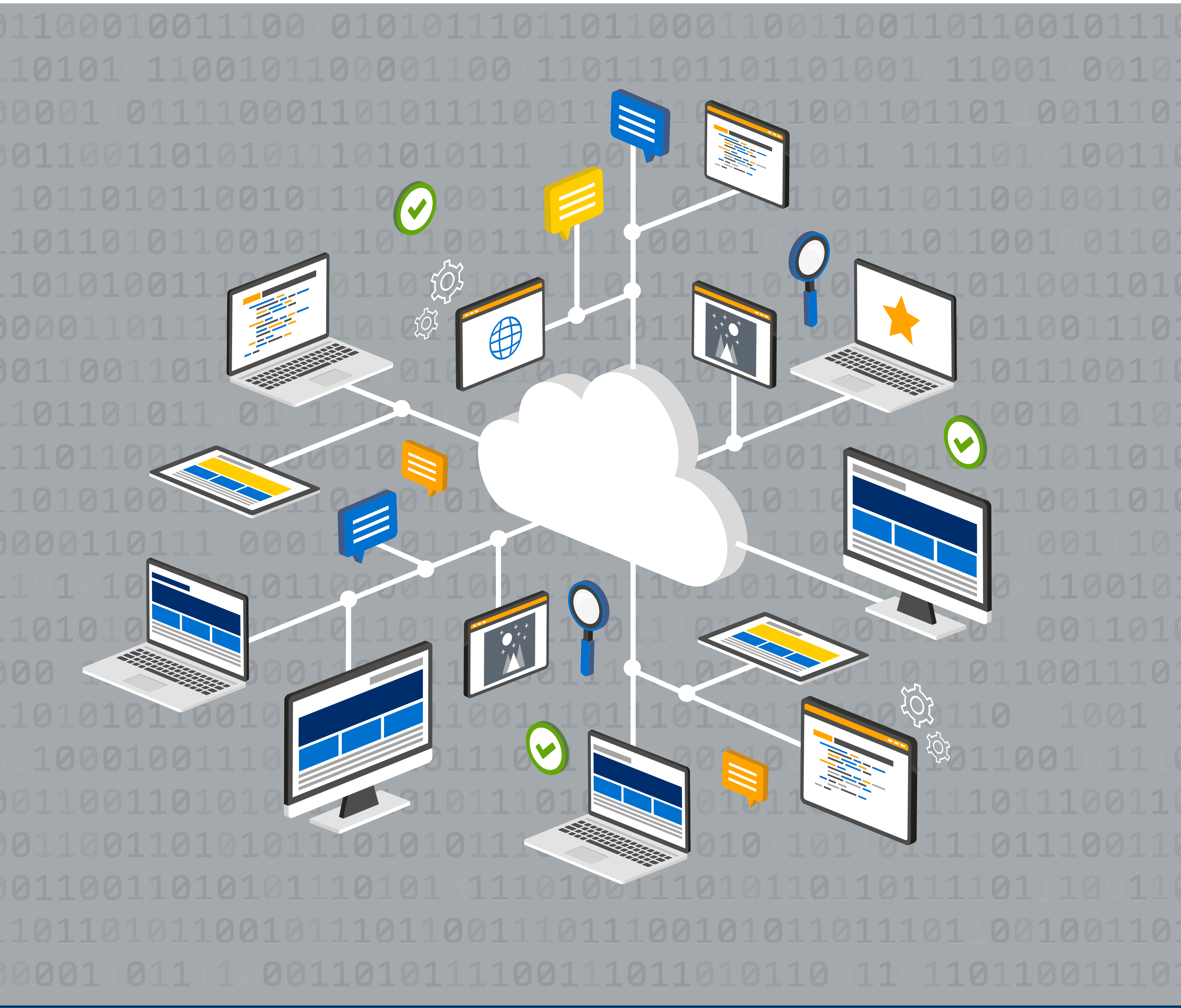
LogRhythm®
The Security Intelligence Company



Detecting Compromised Systems
Analysing the top eight indicators of threat traffic

A Randy Franklin Smith white paper commissioned by LogRhythm

Table of contents

Introduction

The external threat is one of the most high-profile risks that organisations face. Representing more than 80 percent of attacks today, external attackers look to take advantage of network and user weaknesses via malware, phishing, and advanced persistent threats (APT).

Command and control (C2) malware (e.g., ransomware and Trojans) as well as malware designed to exfiltrate data are two of the three most common threats[1]. These processes find their way into your organisation via email phishing scams or compromised websites that are laden with malicious code and are designed to infect your endpoints.

Servers and end-user devices are nearly equally desired targets[1], making just about any endpoint a target. With a majority (60 percent) of organisations focusing their endpoint security strategy on securing data rather than devices[2], malware and other malicious processes somehow inevitably gain a foothold within your organisation.

After ransomware or advanced persistent threat (APT) malware embeds and activates itself on an endpoint, the malware first attempts to connect externally to a C2 server to obtain instructions. Catching this attempt as early as possible is optimal, but even finding it within the exfiltration phase of an attack provides value to the organisation.

Key indicators of a compromise can be found by analysing the network traffic from outbound connections—specifically, traffic coming from an endpoint on your internal network and connecting through your firewall to something on the internet. Focusing on this threat traffic will give your organisation visibility into early indicators of a potential threat.

The goal is to detect a compromised endpoint. Endpoint security solutions certainly assist with this aim, but whether you have such technology deployed or not, the analysis of anomalous network traffic is critical to detecting ongoing compromised systems.

So, what are the best ways to identify a compromise from network traffic alone?

In this paper, we review eight sets of network-related traffic, from the potentially suspicious to the downright malicious and discuss how you can use each to detect a compromised system.

Starting with the right tools

To use traffic analysis to detect compromised systems on your network, you need a network analysis tool and a network tap or switch that supports port mirroring. Because the focus is largely on outbound traffic, analysis can take place within your demilitarised zone (DMZ) or just inside your firewall, as appropriate.

Detection made easy

The process of investigating network traffic for possible signs of compromise requires special tools, and most IT pros haven't armed themselves with a network analysis and forensics tool.

Fortunately, LogRhythm's Network Monitor Freemium, a free solution, provides the Layer 2-7 visibility that you need to recognise suspicious network traffic. This solution can aid in detection of and investigation into unwanted and unauthorised applications and their resulting traffic.

Look in this paper for insights from LogRhythm and examples of how to best use Network Monitor Freemium to detect threat traffic.

You can obtain LogRhythm Network Monitor Freemium at the link below:

logrhythm.com/freemium

[1] Verizon, Data Breach Investigations Report (2016)

[2] Ponemon, State of the Endpoint Report (2016)

The top eight indicators of compromise in network threat traffic

To effectively detect a compromised system, there are eight types of network traffic that you should monitor. We'll cover those here.

1. Reputation of destination IPs and domains

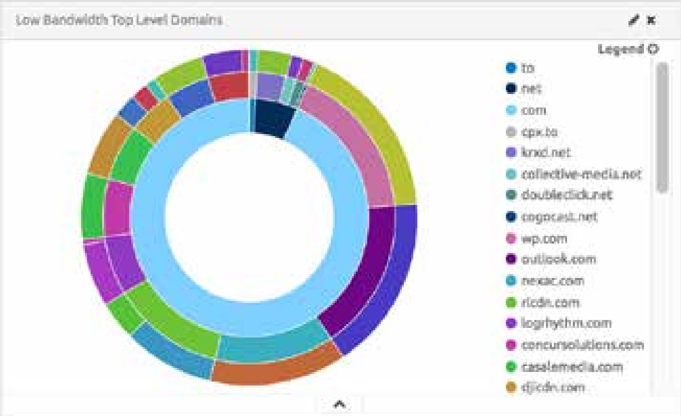
The easiest way to detect inappropriate traffic is by looking at where the traffic is going. Any domains or IP addresses that are on blacklists or that have low reputations are prime candidates.

Outbound traffic data, along with destination IP addresses or domains, can be forwarded to your security information and event management (SIEM) solution, automating the process of validating the reputation of each destination IP address or domain. (Most SIEM solutions can integrate with outside services such as a blacklist or reputation list providers.)

Another way to spot potential threat traffic is to look at anomalous destination domains or IP addresses. Those that are new, as well as lower-volume outliers, can indicate suspicious outbound traffic.

LogRhythm Insights: Outlier traffic

Having visibility into where traffic is going—at both a top- and second-level domain perspective—helps you better understand what is and isn't "normal" for your network. But finding outliers (which, by definition, aren't normal) is an even tougher prospect.



You can configure LogRhythm Network Monitor Freemium's dashboards to show low-bandwidth traffic by top-level domain (shown in this figure as the innermost ring), as well as second-level domains and subdomains (shown as the middle and outer rings, respectively). Metadata, including bandwidth consumption, time of use, and dozens of other pieces of information (on a per-packet or per-flow basis), all provide needed context around the specific nature of suspicious outlier traffic.

2. Unrecognised protocols

Every port that is used in network communications generally identifies which application is responsible for the traffic. Because many instances of malware communicate by using a proprietary application or service, the traffic can be sent over a completely unknown port. This analysis is quite simple, requiring observation only of traffic that originates from endpoints outside the normally allowed ports. (You can determined the allowed ports by referencing your firewall rules.)

You might wonder why you should bother analysing traffic outside of what the firewall allows. The effort might seem a bit counterintuitive, as that traffic isn't allowed anyway. But remember: A compromised machine at least attempts to communicate in its programmed manner. So looking for communication attempts from endpoints can help to identify compromised systems, even when those attempts are unsuccessful.

Another instance of anomalous use of protocols can be Secure Sockets Layer (SSL) traffic that bypasses your SSL proxy. Malware isn't the slightest bit interested in your endpoints' SSL proxy settings, so it often performs its intended communications without the help of an otherwise established proxy server. SSL traffic that originates from an endpoint and establishes a session with an external host—all without the use of your designated SSL proxy—should be considered suspicious.

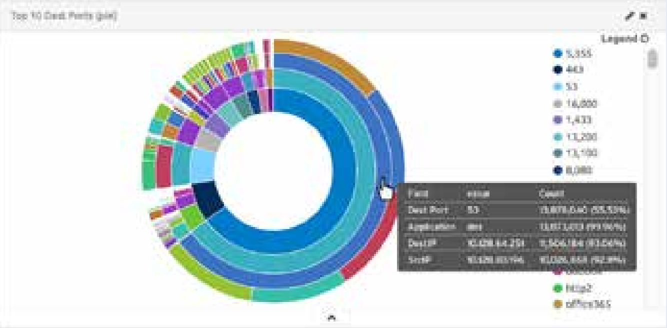
3. DNS queries from clients on your network

One method that external attackers use to compromise a system is to replace DNS settings to point to servers of their own, thereby controlling with which servers the compromised system communicates. This approach enables the attacker to cause data to be sent out to incorrect servers, redirect the user to further malware-infested sites, and so on. This method is a powerful, yet simple, way to establish a permanent foothold on an endpoint.

When reviewing outbound DNS traffic, you should always see a source IP of an internal DNS server, as endpoints do not usually send DNS queries directly to a DNS server on the internet. For example, in a Windows environment running Active Directory, workstations that are part of a domain need DNS to point to domain controllers (DCs) that host DNS services to first find and authenticate to the domain and then to find an address on the internet. Because workstations point to DCs, the normal DNS traffic to the internet should originate from the internal DNS servers. Any DNS queries that come directly from an endpoint are a potential indicator that the endpoint's DNS settings have been hijacked.

LogRhythm Insights: Finding DNS compromise

DNS is a chatty protocol and it likely dominates your network traffic from a count perspective. DNS also should have very few source IP addresses, as the protocol should be centralised with a few internal DNS servers making queries to the outside world and providing answers to endpoints, as shown in this figure.



Field	Value	Count
Dest Port	53	13,878,040 (55.53%)
Application	dns	13,873,073 (99.96%)
DestIP	10.128.64.251	11,506,184 (83.04%)
SrcIP	10.128.83.194	10,026,858 (92.8%)

Because of the limited number of DNS query source IP addresses, this indicator is one of the easiest to search for. With a simple filter that looks for DNS traffic without a source IP that matches one of the few servers that host DNS internally, you can quickly find any endpoints that might be compromised.

In addition, look at the query detail to see abnormal domain requests to domains that use autogenerated names (e.g., qhfieysn.info), which can indicate potential risk. Malware uses domain-generated algorithms to create large numbers of domains that act as C2 servers, to avoid detection by reputation and blacklist providers.

4. Suspect traffic patterns

To spot threat traffic, you can take advantage of malware's dependence on a few specific uses of outbound communication. Malware often calls home to a C2 server to obtain its next set of commands. Or malware can be designed to exfiltrate data (or everything typed by the user) in the hopes of obtaining credentials. These patterns generate either abnormal communications to high ports (e.g., TCP 6667) or unusual amounts of traffic over "allowed" ports.

LogRhythm Insight: ICMP tunneling

Some traffic patterns can be identified as a potential threat based only on the amount of traffic being sent. Take, for example, Internet Control Mapping Protocol (ICMP) traffic. This lower-layer protocol (unlike the browser, DNS, and other application Layer 7 traffic discussed previously) uses no ports. Tools such as Ping and TRACERT use this protocol to test connectivity.

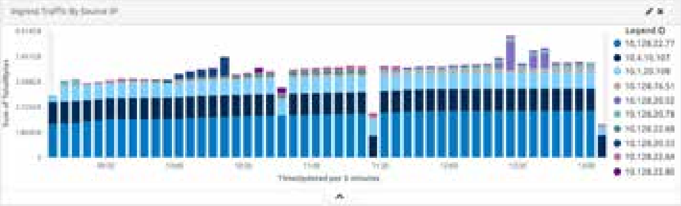
Some malware leverages ICMP tunneling (in which data is injected into an echo request packet, is sent via ICMP, and obtains a response sent in the same manner). Because of the lack of ports, separating out threat traffic from normal traffic is a bit more difficult. Deep packet inspection is required; look at the packet size, rather than just the protocol and port, to identify potential threat traffic. Normal ICMP packet sizes are 48 bytes, but when ICMP tunneling is used, packet sizes range well beyond the norm.

Normal outbound traffic generally falls into a relatively small number of destination ports (e.g., 80 and 443 for web traffic, 25 for SMTP, 53 for DNS) and a relatively predictable amount of traffic for a given type of application over time. Every destination port corresponds to an application, so you have a number of ways to spot suspect traffic:

- Odd combinations of outbound protocols and time of day: A spike in outbound HTTP traffic at 3:00 a.m. on a Saturday is worth investigating in most companies where you wouldn't expect a user or automated process to be active at that time.
- Application and protocol mismatches: Malware often uses known good ports, but a deeper dive into the traffic patterns might show that the application doesn't match what is expected (more on this point later).
- Bandwidth imbalance: For specific activities, such as web browsing, the majority of the traffic is inbound. Spotting an imbalance, in which an unexpected amount of traffic is outbound, can indicate a potential exfiltration scenario.

LogRhythm Insights: Outbound traffic by source IP

One of the easiest ways to spot abnormal traffic patterns is to look at which systems are sending which traffic outbound over time. By viewing this data, you can identify both normal patterns (such as the approximately 4 GB of outbound traffic every hour by the same endpoint in this figure) and those that might be suspect (such as a spike in traffic after hours by a machine normally not in use at that time of day).

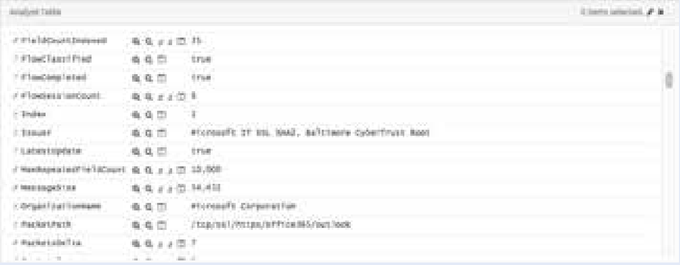


With web-browsing traffic being a prime medium for both C2 and exfiltration traffic, and with the ability to secure that communication via SSL, how are you supposed to monitor HTTPS traffic? You can place an SSL decryptor in front of your network analysis tool. A number of well-known vendors have decryptor solutions, each with a slightly different method of decryption. For example, some require the use of a specific certificate to encrypt in order to facilitate man-in-the-middle decryption. None are 100 percent effective, so some encrypted traffic goes unanalysed.

You can also look at using an SSL proxy, in which the traffic is encrypted not from the endpoint to the destination host, but at the proxy to the destination host, leaving the traffic between the endpoint and the SSL proxy captured as plaintext.

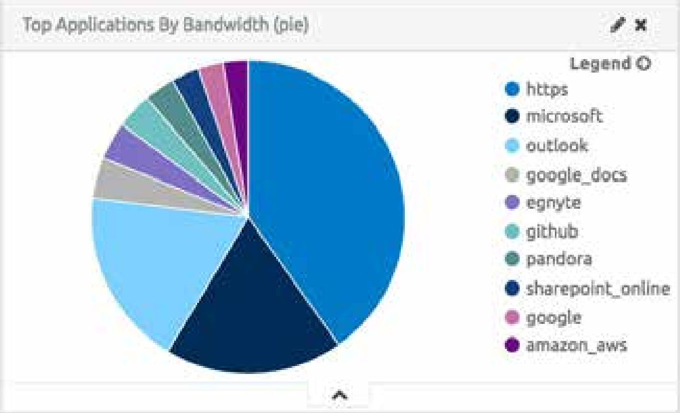
LogRhythm Insights: Analysing HTTPS traffic

Although you might think HTTPS traffic is a black box, the reality is that even when HTTPS is used, a wealth of information in the server certificate (shown here) is still available prior to the decryption of the session. This information becomes helpful during any kind of investigative activity, because you can establish the credibility of the certificate by looking at whether it is self-signed (a frequently used tactic with malware), who the certificate authority is, whether the server matches the Fully Qualified Domain Name (FQDN) in the certificate, and where the traffic is going.



```
Analyst Table
FieldCountIndexed        75
FlowClassified           true
FlowCompleted            true
FlowSessionCount         8
Index                    1
Issuer                   Microsoft IT SSL SHA2, Baltimore CyberTrust Root
LatestUpdate             true
MaxRepeatedFieldCount    10,000
MessageSize              14,432
OrganizationName         Microsoft Corporation
PacketPath               /tcp/ssl/https/office365/outlook
PacketsDelta             7
```

Additional information is available simply by looking at the service with which the HTTPS session is being established. Taking the example here, the analysis shows sanctioned applications such as Microsoft Office 365, Amazon Web Services, and Google. Should a service in the list not be one that the organisation uses (e.g., egnyte or sharepoint_online), that service might require further scrutiny.



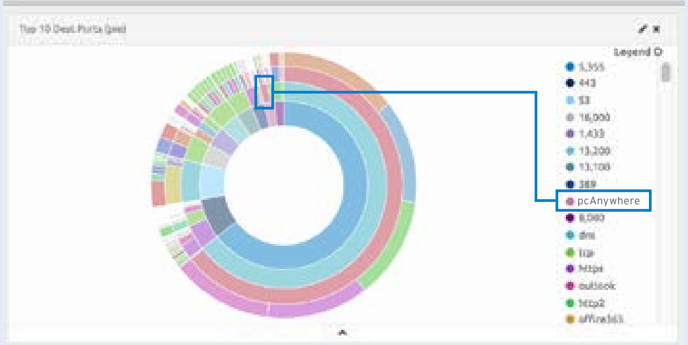
5. Masquerading protocols

One of the sneakiest ways in which malware tries to fly under the radar is by hijacking a known protocol's port. Think about it: Just because traffic over port 53 (which is associated with DNS and normally allowed through a firewall) is being sent externally, there is no guarantee (or requirement) that the data that is sent over that port is actually DNS traffic. If embedded malware communicates to its C2 server over port 53 (and your firewalls allow any endpoint to communicate with external hosts over port 53), then the malware will be successful in its communication attempt.

Identifying this kind of threat traffic requires looking past the assumed purpose for the port (DNS, in the previous example) and analysing to the application protocol to determine whether it matches the structure of traffic that is expected for a given combination of protocol, port, and application. Consider this triple combination, because you cannot always assume that a protocol, such as HTTPS, will run only on port 443. You can establish web-browsing sessions over any port. You also cannot assume that port 443 traffic is always HTTPS or web-browsing traffic. Many remote-session vendors communicate over port 443 to avoid requiring special firewall rules to function.

LogRhythm Insights: Looking for threats in the exceptions

By and large, most traffic over a well-known and well-used protocol is appropriate traffic. Malware masquerading as traffic of another allowed protocol is the exception. If you find that 99.96 percent of your DNS traffic adheres to expected payload patterns, you might think that everything is fine. But looking at that .04 percent of the traffic will yield suspect and potentially malicious threat traffic. For example, by eliminating all DNS traffic over port 53, you can expose all the remaining traffic that uses that port. And if, as is in the figure here, you find some pcAnywhere remote-session traffic running on the port, that traffic is highly suspect.



So, plan for part of your diligence to include inspection of traffic to ensure that it aligns with the applications that are normally used over a given port.

6. Known signatures

Malware, in general, leaves a trail of consistent traffic patterns that can be used to generate a database of known signatures. Deep packet inspection can then compare traffic with the signature database to identify threat traffic. However, signature-based detection tends to be less effective for two reasons:

1. You need to remain constantly up to date on the traffic patterns that malware uses—likely getting that information from a proprietary intelligence provider.
2. There are literally tens of thousands of new variants of malware are released daily.

7. Prohibited protocols

A number of protocols are used as part of your network but are prohibited for general use by internal endpoints. Protocols such as SMTP, SSH, VPN, RPC, and IRC are either used in specific IT-sanctioned cases or are simply not allowed. Analysing the use of prohibited protocols (including, but not limited to, the previous list) can help to indicate threat traffic.

To properly identify threat traffic via this kind of protocol use, you must address two issues. The first is: Which endpoints should or normally do communicate using a given protocol?

Take SMTP, for example: Unless someone in your organisation is using an internal email client that is configured to use POP3 and SMTP and communicate directly with the email provider, you don't expect to see endpoint-based SMTP traffic. The same applies to your server running enterprise applications. Take your on-premises Microsoft Exchange environment. An Exchange server that provides a Transport server role is expected to interact over SMTP with the outside world, but an Exchange server that provides only a Mailbox server role is not.

The second issue is: How much traffic is "normal" over a given protocol? Baseline the bandwidth use of each protocol's traffic and set up alerts for when traffic amounts fall outside those established norms.

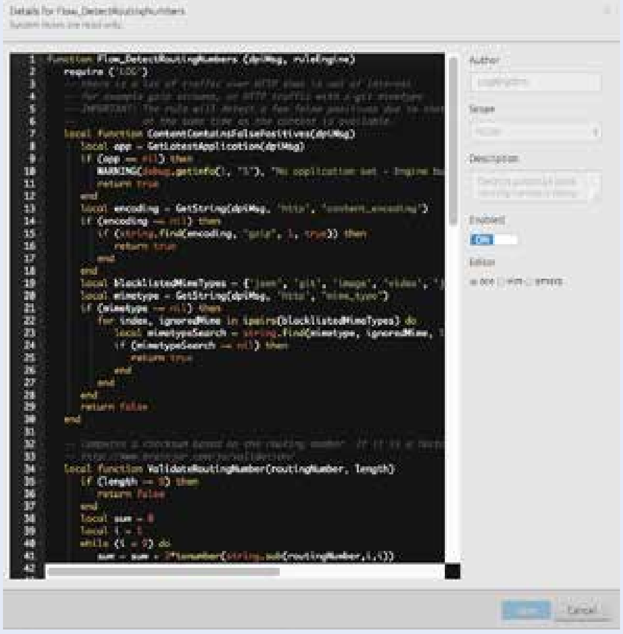
8. DLP indicators

In a simplistic way, data loss prevention (DLP) solutions focus their efforts on classifying data patterns (such as a U.S. social security number [SSN] pattern of ###-##-####), reviewing traffic for any instances of that pattern, and blocking the copy of files, sending of attachments, and so on when that pattern is identified.

Even if you already have DLP in place, you can erect another layer of defense by reviewing traffic for those same patterns—looking within the data payloads for regular expressions and keywords that are relevant to proprietary information of your organisation—to identify when suspect activity is occurring. Use of an SSL decryptor or SSL proxy (with your network monitor tap place before the proxy) might be necessary for the greatest possible visibility.

LogRhythm Insights: Building out rules to search for PII

Analysis of traffic to find personally identifiable information (PII) is actually a simple task, because you likely know which kinds of protected information exist within your organisation, and because many organisations store the same types of information (e.g., SSNs, banking information, credit card numbers). LogRhythm's Network Monitor Freemium uses built-in rules, based on the Lua scripting language, to perform deep packet analysis, looking for specific patterns such as bank routing numbers, as shown in this figure.


Details for Flow_DetectRoutingNumbers
System Rules are read only.
Author
Scope
Description
Enabled
ON
Editor
ace
vim
emacs
Save
Cancel

```
function Flow_DetectRoutingNumbers (dpiMsg, ruleEngine)
    require ('LOG')
    -- There is a lot of traffic over HTTP that is not of interest
    -- for example gzip streams, or HTTP traffic with a gif mimetype
    -- IMPORTANT! The rule will detect a few false positives due to the
    --     at the same time as the content is available
    local function ContentContainsFalsePositives(dpiMsg)
        local app = GetLatestApplication(dpiMsg)
        if (app == nil) then
            WARNING(debug.getinfo(1, "S"), "No application set - Engine b
            return true
        end
        local encoding = GetString(dpiMsg, 'http', 'content_encoding')
        if (encoding ~= nil) then
            if (string.find(encoding, 'gzip', 1, true)) then
                return true
            end
        end
        local blocklistedMimeTypes = {'json', 'gif', 'image', 'video', '
        local mimetype = GetString(dpiMsg, 'http', 'mime_type')
        if (mimetype ~= nil) then
            for index, ignoredMime in ipairs(blocklistedMimeTypes) do
                local mimetypeSearch = string.find(mimetype, ignoredMime, 1
                if (mimetypeSearch ~= nil) then
                    return true
                end
            end
        end
        return false
    end

    -- Computes a checksum based on the routing number. If it is a techn
    -- http://www.brainjar.com/js/validation/
    local function ValidateRoutingNumber(routingNumber, length)
        if (length ~= 9) then
            return false
        end
        local sum = 0
        local i = 1
        while (i < 9) do
            sum = sum + 3*tonumber(string.sub(routingNumber,i,i))
```

By using these rules and by building your own, you can quickly identify suspect transfers of these types of data. Be aware that DPI for PII can result in numerous false positives, as data that matches PII patterns is present on your network. Some fine-tuning, including some of the other indicators in this paper as well as spotting PII, is required to find not just the suspicious, but the malicious. Note that you can easily clone the rule and make a custom version based on your network traffic, removing any false positives that you have evaluated and cleared.

Find the threat traffic: Find the compromised system

Malware cannot exist without revealing itself somehow. One requirement that malware just cannot get around is the need to communicate. Whether by trying to spread itself laterally within your organisation or by attempting to obtain updated commands from a C2 server, malware must rear its ugly head in the form of network traffic.

By putting a network-monitoring solution in place and analysing your network's traffic using the eight indicators of threat traffic, you can quickly determine whether compromised systems exist on your network, and if so, where.

Remember: Although this paper spells out the top eight indicators of a compromised system, you need to fine-tune your dashboards, filters, scripts, and so on to eliminate the noise of false positives, honing in on the traffic that indicates the existence of active malware.

Keep in mind that malware is somewhat of a living, evolving threat—one that you need to stay abreast of and respond to accordingly. So although these top eight indicators serve as a solid foundation of what to look for, be cognisant that in the future, you'll surely need to change which traffic you analyse and how you do so.

About LogRhythm

LogRhythm, a leader in security intelligence and analytics, empowers organisations around the globe to rapidly detect, respond to and neutralise damaging cyber threats. The company's patented award-winning platform uniquely unifies next-generation SIEM, log management, network and endpoint monitoring, and advanced security analytics. In addition to protecting customers from the risks associated with cyber threats, LogRhythm provides unparalleled compliance automation and assurance, and enhanced IT intelligence.

LogRhythm is consistently recognised as a market leader. The company has been positioned as a Leader in Gartner's SIEM Magic Quadrant report for five consecutive years, named a "Champion" in Info-Tech Research Group's 2014-15 SIEM Vendor Landscape report, received SC Labs "Recommended" 5-Star rating for SIEM and UTM for 2016.

LogRhythm is headquartered in Boulder, Colorado, with operations throughout North and South America, Europe and the Asia Pacific region.

Download LogRhythm's Network Monitor Freemium at www.logrhythm.com/freemium

About Randy Franklin Smith

Randy Franklin Smith is an internationally recognised 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 organisations. Randy is also a Microsoft Security Most Valuable Professional.

DISCLAIMER: Monterey Technology Group, Inc. and LogRhythm make no claim that use of this white paper will assure a successful outcome. Readers use all information within this document at their own risk.