A DNS Server on the Box

With this month’s patch tuesday update to Network Box 5, we’re releasing a DNS recursive resolver to the Network Box 5 platform. Based on an enterprise-grade DNS server, this allows the box itself to recursively resolve DNS queries itself (without relying on external recursive resolvers, and able to operate off the root level DNS servers themselves). This service can also be offered to workstations and servers on the LAN/DMZ (so such machines can use the Network Box at the gateway as their specified DNS server).

Building on that, we’ve also released this month an optional network-dns-server security module. This security module provides the capability to configure DNS zones as either master or slave, on the Network Box itself. The Network Box will serve queries to those zones in an authoritative manner.

Cloud DNS Backup

In November 2014, we announced the launch of our Cloud Mail Backup service and since then have been rolling this out globally to our Network Box 5 customer base. Since then, we’ve also added a country-level cloud backup centre in Germany (which provides the option to keep the eMail within Germany for those European customers with privacy concerns).

As we promised then, Cloud Mail Backup was just the first of many cloud services to be launched in the coming months, and this month we’re pleased to announce the launch of our second cloud service - Cloud DNS Backup.
DNS Servers in the Cloud

To further extend the capabilities of this system, we’re also releasing Cloud DNS Backup. This allows customers to use Network Box’s extensive network of cloud DNS servers to provide backup DNS in the cloud. By using such a cloud based network, service reliability can be improved and DNS queries can be responded to, from the cloud, even if the Internet link to the master DNS server in the LAN/DMZ is down.

As with Cloud Mail Backup, the customer has full control over which Cloud DNS servers in which regions and countries, should be used to provide the service.

At any given time, the current status of Cloud DNS Backup for a particular customer can be checked in ‘Box Office’, by looking at the owner account, ‘Cloud DNS Backup Domains’ section. (This is typically updated once a minute.)

This is a FREE SERVICE provided to all Network Box 5 customers subscribing to packages including our network-dns-server security module.

Over the coming months, we plan to continue to release more of these optional Cloud services. The intent of this work is to offer improved security, functionality, and business continuity, by leveraging cloud technologies, to improve on what we can offer Network Box clients around the world.

Conditions

- Clients will have full control over which of their domains will use the service.
- Clients will have full control over which cloud backup servers will be used for their domains.
- In general, the DNS system operates with load-balanced round-robin behaviour. The selection of DNS server to be used to resolve a particular query is often random and unpredictable. It is important to ensure that all listed servers are authoritative and correctly configured for the domains they are serving.
- Only the DNS records themselves, as well as statistical logs, will be stored on the Cloud DNS Backup servers.
- Should a particular Cloud DNS Backup service be unavailable for any reason, it will not respond and the client will choose an alternate server to query (as per the standard mechanics of the DNS protocol).
For many years now, we’ve become accustomed to seeing application vulnerabilities, with some of these being remotely exploitable and the worst being in network services. We’ve learnt how to categorize and prioritize these threats, and have good response procedures in place to mitigate and respond to them.

However, in the past year, we’ve seen a number of core library level vulnerabilities affecting multiple applications in multiple different ways. This is a worrying trend where the application itself is not targeted, but instead one of the system core libraries it uses has a vulnerability that can be exploited through the application. Such exploits are often remotely achievable.

Examples of this include the bash shellshock vulnerability, heartbleed and other issues in the core openssl library, and most recently the glibc ghost vulnerability.

In the face of these new types of vulnerabilities, as an industry, we’ve got to learn new techniques to handle them from a security response point of view.

Some things to bear in mind include:

• In the case of application vulnerabilities, there is usually only one exploit vector. But, in the case of core library vulnerabilities, each application that uses that core library may have many possible exploit vectors.

• Detecting core library exploit code, at the network level, with so many possible exploit vectors, is not at all straightforward (especially compared to application vulnerabilities). A single core library exploit may require dozens of signatures to cover even just the common exploit vectors.

• Once an application is patched to address a particular vulnerability, the application can simply be restarted to install the fix (which is commonly even part of the automatic patch application script). However, for core library vulnerabilities, after applying the patch you must manually identify and restart each affected service.

• Services using the Unix/Linux fork-and-exec approach won’t automatically pick up a newly patched core library and will continue to fork copies of vulnerable code until restarted.

• The safest approach is a complete system reboot, but that can be obtrusive and impact service delivery.

Network Box Security Response, as are others in our industry, are working hard to introduce security technologies and revise our response procedures to address these and other such vulnerabilities. In the case of known core library exploits that could affect the Network Box itself, we’ll remotely patch and restart affected services. In the case of vulnerabilities that affect protected customer systems, we work with our industry partners to identify possible exploit vectors and address those appropriately.
Most of the literature I’ve found on this topic still refers to classify and remediate in these terms. In the meantime though, I’ve noticed that some companies are using these terms in a very different way and making lots of strides in marketing their products with scare tactics, using big terms.

Many years ago, we used to install IDS, which would produce hundreds of pages of reports and, once each day, we’d pore over those pages to find correlations and identify possible threats. After that, we’d make configuration changes, mostly to the firewall, to block those threats.

With the passing of time, we’ve built tools that no longer require such human intervention, for many reasons. Aside from being inconvenient and demeaning for someone to spend his or her life poring over hundreds of pages of logs on a daily basis, it’s also incredibly inefficient and easily plays into the hands of hackers. With modern hacking counting on automated tools that change the threat at a speed beyond human comprehension (let alone capability), relying on human analysis is, to say the least, obsolete (I’d venture further and call it obscene at this point).

As such, we’ve created tools that identify threats in true real time and block them, automatically. These are called IPS – intrusion PREVENTION systems. Prevention is key here; you want to block the attempt at intrusion as it starts. You want to change the paradigm from having a camera that gives you a picture of the thief, to a guard that blocks the thief at the doorstep. If a packet can be malicious, block it.
An evolution of this concept consists of integrating the IPS with the firewall; allowing the firewall to do most of the work as it’s faster and much more accurate. Then scan the traffic that passes through with the IPS and, if the IPS determines that a packet is a threat, tell the firewall to drop that connection, so no further threat packets can pass through from that sender. This is what many are calling the NextGen Firewalls, which, incidentally, are old news now. Network Box has had this concept since 2000; only relatively recently has the industry caught up with it and given it a formal name.

Since this approach makes a lot of sense, you’d expect that the entire security industry would move in that direction and abandon the concept of IDS. In a way though, IDS still has merit within a LAN; it can identify malicious traffic within the LAN and provide alerts on local activity.

Enter the concept of ‘classify and remediate’. I’ve run into this more than once lately. There are companies that make a fortune telling their customers that they’ll monitor the traffic and alert them if something’s trying to come in. And I’ve even seen such “alerts”. One of them went like this “IP address xyz is sending a PHP exploit to your web server; please make sure your server is patched”.

Seriously? That’s their take on security? Please, make sure your server is patched?

I was (and still am) flabbergasted at such daring. I’m not talking about small companies who are trying to make a living creating unneeded panic. I’m talking about large security companies that instead of blocking the traffic and protecting their customers using an IPS, use an IDS to alert and ask the customer to ‘remediate’. I feel as though we’ve gone back 15 years and found a way to market an old concept. We’re using the concept of classify and remediate as a viable protection from Internet threats.

Well, I’ve news for all of you attempting to do this – it does NOT work, and it’s the reason why so many companies are falling victim of hackers left and right. You cannot “remediate”; you need to block before the issue even starts. Having the ability to identify a threat floating inside a LAN is important. But if you truly want to be protected from Internet threats, stop thinking in terms of classify and remediate, and start thinking in terms of active protection, real time blocks, a barricade against each and every threat that the Internet is going to throw at you.

Knowing that IP xyz is scanning your ports is irrelevant and unnecessary. It makes for a nice graph you can present to the board of directors, but nothing more. What you really need to have is a system that will BLOCK such threats; so that IP xyz can knock at your door as many times as it wants; if you’ve configured things properly, the door will never open and those “threats” will continue to be caught, and blocked at the edge of your network, unfailingly, like only an automated system can do.

If you’re still relying on human intervention for your protection because these companies have convinced you that it’s the way modern security works, it’s time you look around and get yourself some REAL security.
Transport Layer Security (TLS) and its predecessor Secure Sockets Layer (SSL) are cryptographic protocols. They are designed to permit the authentication of clients to servers, and vice-versa, as well as protect the communications from such threats as eavesdropping, tampering, and replay. The protocols have been around for more than 10 years now, and are seeing near ubiquitous usage as the protocol of choice for secure communications of many sorts.

Certificate Verification

Now, when a client connects to a server talking TLS/SSL, the server can provide its own certificate to the client.

When the client receives this certificate, it can verify the authenticity by checking against a list of trusted certificate authorities it maintains. By using the certificate authority’s public key (inside a certificate), the client can verify the signature on the certificate presented by the server. You can see that for this to work, the certificate authority who signed the server’s certificate must also be trusted by the client (ie; in the list of trusted certificate authorities installed on that client).

Once the certificate has been verified, key exchange can be safely performed using the public key inside the certificate (as only the server has the matching private key used to decrypt).

Of course, the above is a vastly simplified overview of a complex process, but you should be able to see the basic mechanics of how this works. The key points to take away from this are:

1. Data encrypted with a public key can only be decrypted with the matching private key. This is asymmetric key cryptography, and works both ways public->private and private->public.
2. The certificate contains a public key, as well as matching identification information for the owner.
3. The certificate is signed by the private key of a certificate authority trusted by both the client and the server.
4. The client can verify the signature using the public key of the trusted certificate authority.
5. The client can send data securely to the server using the verified public key of the server (as only the server has the matching private key used to decrypt).

The Certificate

At the core of TLS/SSL is the certificate. This is based on RSA asymmetric key cryptography, where a key pair (public and private) is generated such that one key is used to encrypt while the other is used to decrypt. So, for example, we can encrypt some text with the public key, and then only the holder of the private key can decrypt it, or vice-versa. Such asymmetric key systems can be used for encryption/decryption as well as signing/verifying (I sign with one key, then you can verify with the other).

To generate a certificate, two entities are involved.

1. Firstly, the owner of the certificate generates a random public+private key pair, and then puts the public key plus his identifying information into a certificate request, and sends it to a trusted certificate authority.
2. Secondly, the certificate authority generates its own public+private key pair (or more commonly uses one previously generated) and then uses its own private key to sign the owner's public key in the certificate request.

The result of this process is a certificate containing the owner's identifying information and public key, signed with the private key of the trusted certificate authority.

In addition, the trusted certificate authority puts its own public key into a certificate and typically self-signs it with it’s own private key (or in some cases gets it signed by an another upstream trusted certificate authority), then publishes that certificate.
**Man In the Middle**

So, say that as an organizational policy, we want to be able to inspect TLS/SSL protected traffic. We want to ensure that TLS/SSL protected web sites being visited over the HTTPS protocol conform to organizational policy, and that the files transferred do not contain malware. We also want to ensure that proper validation of the TLS/SSL certificate is performed, and that protocol exploits are protected against.

The way to do this is with a so-called man-in-the-middle interception.

The idea is that when the client connects to the server, that connection is intercepted by the proxy. The proxy then maintains two separate TLS/SSL connections - one from the client to proxy and the other from the proxy to the server. This is performed as follows:

1. The proxy itself has its own certificate (with public and private key) and this certificate is installed as a trusted certificate authority into the client.
2. The proxy negotiates TLS/SSL with the server, and validates and ensures the correctness of this connection.
3. The proxy takes the server certificate, replaces the public key with one of its own, and then self-signs that certificate with its own certificate authority private key.
4. The proxy uses this modified certificate to negotiate a TLS/SSL connection with the client.

When talking to the server the proxy sees the server’s certificate signed by a certificate authority trusted by both the server and the proxy. The server’s public key (contained in that certificate) can be used to securely communicate with the server. But, the client sees the server’s certificate signed by the intercepting proxy with the proxy’s public key, and the proxy’s public key (contained in that modified certificate) can be used to securely communicate with the proxy.

With such an arrangement, the proxy decrypts the traffic from the client, examines it for policy enforcement, then encrypts it and forwards it on to the server. Return traffic from the server can be decrypted by the proxy, examined for policy enforcement, then re-encrypted and forwarded on to the client. The communications client-proxy and proxy-server are secure, but the traffic is subject to policy enforcement.

**Strengths and Weaknesses**

Such an arrangement removed the weak link (the end-user policy decision) from the TLS/SSL protocol, by allowing policy enforcement to be made by administrators at the network gateway. It also allows for protocol upgrading, and protection against exploit of browser and other client-side vulnerabilities. A good example of this is the recent FREAK vulnerability - Network Box 5 SSL proxy users have been protected against exploit of that from day #1, as the Network Box 5 SSL proxy already enforces secure cryptographic cipher selection at the gateway (irrespective of the vulnerability at the client).

However, there are two main areas such an arrangement may not work:

1. **Client-Side Certificates.** The TLS/SSL protocol allows for client-side certificates. If these are used, the traffic cannot be intercepted (as the client certificate provided to the server could not be modified without access to the server’s trusted certificate authority list - which is not practical).

2. **Certificate Pinning.** Some client-side applications using TLS/SSL perform an extra validation step which is to verify the identity of the certificate authority used to sign the server’s certificate. For example requiring the certificates presented by server X must be signed by authority Y. Such traffic cannot be intercepted (as the client-side application would object to the certificate authority used by the intercepting proxy).

In such cases, policy rules are normally put in place, at the proxy, to bypass such sites from TLS/SSL interception.

The other restriction is that the certificate authority used by the intercepting proxy must be trusted by the clients (normally meaning that the proxy’s SSL certificate must be installed into the trusted certificate store of the client). We are often asked why this is a requirement, and the answer is simple - if it wasn’t a requirement, then we could simply and easily intercept, monitor and modify TLS/SSL communications transparently to the client and the security and integrity of the world’s financial networks would be destroyed.
Proxying SSL
part (2 of 2)

We’ve already talked about how Transport Layer Security (TLS) and its predecessor Secure Sockets Layer (SSL) are cryptographic protocols have been used to secure communications for more than 10 years. We’ve also presented how they rely on the certificate, how that certificate is verified, and how man-in-the-middle interception can take place. Now, let’s discuss how Network Box supports SSL in its Network Box 5 product.

SSL as a Filter

The first key concept is that in Network Box 5, TLS/SSL is treated as an encapsulation layer. HTTPS is purely the HTTP protocol encapsulated in an SSL stream. Similarly, SMTPS is SMTP in SSL, POP3S is POP3 in SSL, and IMAP4S is IMAP in SSL. In general, any arbitrary data stream or protocol can be encapsulated in SSL for protection.

Network Box 5 treats SSL interception as a filter on the communication channels. The device can be configured to add or remove SSL from either the input or output sides of a communication. If SSL is chosen to be removed, then the resulting plain text can be analyzed further.

The SSL negotiation is always initiated from the client side, and can be started in one of two ways:

1. Fixed Port (for example tcp/443 for HTTPS). The client starts the negotiation of SSL immediately after the network connection has been established.

2. Command based (for example CONNECT for http proxy, STARTTLS for SMTP, etc). The client first initiates a switch to SSL, using a protocol-level command such as CONNECT or STARTTLS. Then, once the server acknowledges that, both clients assume the connection is now ready for SSL and the client starts the negotiation of SSL.

Client Side vs Server Side

Typically, there are two scenarios where Network Box SSL proxying can be involved:

1. Client Side. Here, clients (typically in the LAN/DMZ) are connecting out to external public SSL servers. In this case, the certificate for the SSL server is not under the control of the Network Box admin. SSL Man In The Middle style interception and proxying must be used to decrypt the traffic.

2. Server Side. Here, clients (typically on the Internet) are connecting in to internal SSL servers (typically in the DMZ). In this case, the certificate for the SSL server is under control of the Network Box admin, but the client policies are not. The server certificate and private key would typically be installed on the Network Box, so that the Network Box SSL proxy can use those to act on behalf of the server and decrypt the incoming SSL sessions.

As you can see, the technologies used for both are very different, so let’s discuss them further individually.
**Client Side SSL**

**Man in the Middle Proxying**

The Network Box 5 SSL Man In the Middle proxy works by intercepting SSL communications between the client and the server. Two separate SSL sessions are maintained - one between the client and the proxy, and the other between the proxy and the server. For more information on the technical aspects of this, please refer to the part 1 article in the April 2015 In The Boxing Ring newsletter.

There are various policy rule points in the Network Box proxy, including:

- The decision on whether to decode the SSL traffic is made shortly after connection time, once the client SSL negotiation message has been received. The options available are (a) bypass the traffic without decode, (b) permit the traffic and decode it, or (c) deny the connection.

- Once the server certificate has been received, it is validated. Checks are made for items such as certificate signing chain, as well as issue and expiry dates. The result of those checks is then available as a policy decision to (a) permit it anyway, or (b) deny the connection.

- When inspecting a communication channel for SSL traffic, at some point a decision is made whether there is actually an SSL negotiation being attempted. In the case where the traffic is not SSL, a policy decision is made whether to (a) bypass the traffic and allow it, or (b) deny the connection. An example here is the HTTP proxy CONNECT statement. That should be used to switch to SSL, and a policy decision can be made if the client does not actually negotiate SSL after the CONNECT statement (as they are most likely trying to bypass firewall policy).

**Protocol Promotion**

One unique feature of the Network Box 5 SSL proxy, that comes from our implementation as a filter, is that the SSL decode mechanism can be combined with other functions such as application identification. We can intercept network communications generically, use the SSL filter to decode the SSL stream, and then apply application identification to that decoded stream. With such a capability, we can detect the full suite of 1,000+ applications, even if they are encrypted in SSL.

Moreover, once we’ve used the filter to decode SSL, we can introduce an identification such as SMTP, HTTP, POP3, etc., use a protocol-specific module for further protection. An example of this would be to intercept all outbound TCP traffic, filter out SSL if detected, then if the application type is detected to be HTTP (originally HTTPS when protected by SSL), transparently switch the traffic through to the webclient module for url policy enforcement, anti-malware scanning, and safe search enforcement.

**Server Side SSL Fronting**

Network Box 5 can be used to ‘front’ SSL communications at the gateway. Rather than deploying a SSL enabled server, instead the SSL certificate and private key can be installed on the Network Box, and the SSL connection terminated there. With this arrangement, the connection from client to proxy is SSL protected. The connection from proxy to server can either be plaintext, or re-encoded as SSL again.

The advantage of SSL fronting is that it allows protection technologies such as WAF and anti-ddos to be deployed even for SSL protected services.

---

**Why Decode SSL?**

So, given the above technologies and capabilities, we come to the core question of why would we want to decode SSL?

The simple answer is that we want to enforce policy control and have the capability to apply protection technologies for traffic protected by SSL, in the same way we do for plaintext traffic. As more and more of the Internet’s communications move to SSL, less and less is subject to policy control and protection, unless we deploy SSL decoding capabilities. An example is google safe search. Sure, we can enforce safe search at the proxy, and it works well in school and home environments. But, google search has now moved to SSL, and without SSL decoding, safe search cannot be enforced at the proxy.

Another reason is that the SSL certificate verification process occurs largely at the client workstation, with the end-user being responsible for the decision as to whether to accept/reject a problem with a particular connection. End-users are typically not trained in the security implications of such decisions, and in general it is better to leave such security policies up to the administrator, to be enforced at the gateway.

---

We hope that you now have an understanding of the SSL/TLS protocol, and how Network Box 5 can filter it out of your network communications, to allow you to apply policy control and other protection technologies to the underlying data stream.
Today, Network Box is proud to release our VPN-5 appliance to provide another option at the lower end, without sacrificing protection capabilities.

The VPN-5 appliance is specifically targeted at remote branch offices requiring VPN connectivity back to head office (in either a hub-spoke or mesh style network environment using VPN or MPLS technology).

The appliance itself is not based on a low power, low memory, low performance, single core CPU like most of its competitors. Instead, it uses a dual core 1.6GHz CPU, 2GB RAM, and offers 16GB (4GB system, 12GB user) flash storage. Coupled with 4x 1Gbps ethernet interfaces, this provides performance to handle even the most high bandwidth remote office connections. All three VPN types are supported (PPTP, SSL and IPSEC), and the device can be fully managed and integrated with the rest of your managed network.

It is intended for firewall and VPN only, and the price is significantly less than S or M series units.

When it comes to large mesh or hub-and-spoke networks, using VPN/MLPS connections to branch offices, the choice of devices to deploy at the remote stations has always been difficult.

Until today, there have been two options:

**Option (a):**
Very cheap and simple devices, with little effective centralised management, poor performance, and limited protection capabilities.

**Option (b):**
More expensive devices, which can be fully integrated to your managed infrastructure, and offer the capability of perform effective protection at the network edges.

Up until now, the Network Box S-series appliances have addressed option (b), and we have been forced to work with less-than-optimal devices to address the requirement for option (a).
This month, we’re proud to release a number of significant enhancements to our VPN reporting system, as accessed via the administrative web portal in Network Box 5. We now offer reporting based on 4 levels, as described below.

Level 0 - Current Status

The current status of connected VPNs can be seen in most VPN devices. This simple display shows you each VPN, when it was connected, and (in some appliances) how long it has been connected for.

Network Box 5 has offered this report for some time now (Network / VPN / Status). It is updated in real-time, and goes a step further than most VPN devices by also showing a real-time updated global map of the status of your VPN connections.

Level 1 - Events

Most VPN devices on the market today offer an event based reporting system. They’ll tell you events such as when a VPN was connected, when it was disconnected, or details of when an error occurred on the link.

This is a pretty basic report. It doesn’t tell you how long a VPN has been up, or even which VPNs were up at a particular date/time, but you can manually process the events to try to determine that for yourself.

Network Box 5 has always tracked these events, but this month we’re releasing a basic report (under Network / VPN / Events) to allow you to see the raw underlying data yourself.

Level 2 - Sessions

A small number of VPN devices available today attempt to correlate the VPN events to track VPN sessions. A session is a continuous period of connectivity of a particular VPN connection. For example, a VPN connecting at 11am on 10th April, and disconnecting at 10am on 13th April, could be said to have a session lasting for duration 2 days 23 hours.

The problem comes when you report on these sessions. You typically want to produce a report saying ‘show me all VPNs connected on 10th April’, and the VPN device will show you any VPN sessions that started on that day. Some will even show you VPN sessions that completed on that day. But, it is rare to find one that will report on sessions established on that day without starting/ending.

Level 3 - In Progress Sessions

To show sessions that were already connected on a specific day (but didn’t start or end on that day) requires tracking the In Progress sessions on a daily basis. This report is the most useful, is surprisingly rare to see offered by VPN devices, and is precisely what we have released for Network Box 5 this month. You can find this new report at Network / VPN / Connections.

The report shows VPN sessions that either started, completed, or were in progress, for a specified time range. For each session, it shows the duration the session was up during that specified time range, as well as the total uptime so far for the session.

Using the above example, and asking for a connection report for time period 11th April, the report would show the connection as having duration 24 hours (the entire day) and uptime 37 hours (from 11am on 10th April to midnight 11th April - the end of the reporting period). Such reports are extremely useful to immediately see VPN connections experiencing downtimes, repeated reconnections, or other such issues.

Conclusions

The three VPN reports are being released this month to all Network Box 5 customers. They are available for all three types of supported VPN (IPSEC, SSL and PPTP), and we hope that you find them useful.

Our work on VPN reporting continues during June, and a second batch of improvements is scheduled to be released in the July patch tuesday. These include:

- Configurability of VPN persistent type. VPNs will be able to be configured as permanent (expected to be up 100% of the time) or ephemeral (disconnecting and connecting at will, with no expected up time).
- Reporting on persistent VPN availability. A report showing the percentage of uptime for each permanent type VPN.
- VPN KPIs will also be released for each of the three main VPN types.
VPN Reporting (part 2)

Last month, we released a number of significant enhancements to our VPN reporting system, accessible via the Network Box 5 administrative web portal. Continuing on from the last set of VPN enhancements, this month we are releasing the following:

- Configurability of VPN persistent type. VPNs can now be configured as permanent (expected to be up 100% of the time) or ephemeral (disconnecting and connecting at will, with no expected up time).
- Reporting on persistent VPN availability is also now released. This is a report showing the percentage of uptime for each permanent type VPN.
- VPN KPIs are also now available for each of the three main VPN types.
- The capability to start/stop/restart VPN tunnels from the administrative web portal.

These improvements have been made for all three types of supported VPN (IPSEC, SSL and PPTP).

Network Box 5 VPN Dashboard
Unification of ACLs and Rules Engine

This month, we are proud to release the results of the unification of our ACLs and Rules engines across all components of our UTM+ platform. This has been a long-term project for us here at Network Box Security Response, and this month marks the culmination as we release the final components of this important system.

The core purpose of a Network Box 5 appliance is to categorise traffic passing through the device, and then to allow policy decisions to be made based on those categories as well as other attributes of the traffic. We don’t block spam, but instead we categorise an eMail message as ‘spam’ and then allow a policy decision to be defined to deny the traffic containing that message. This is a subtle, but important, distinction.

The ACL/Rules engine is the software component that implements those policy decisions. The configuration defines the security ACLs, as well as the rules, and the ACL/Rules engine implements that policy. The issue, up until now, is that our UTM+ appliances contain many components, and each component had its own rules engine. Today’s release unifies all components of the Network Box 5 appliance to use the same core ACL/Rules engine. Previously, the configuration language had been unified. Now, today, the implementation of those rules is unified. This brings important performance and functionality benefits, but most importantly standardises the way we treat ACLs and rules throughout the Network Box 5 system.
ACL Types and wildcarding behaviour

Perhaps the biggest change with the release of the new ACL/Rules engine is the standardisation of ACL types and their wildcarding behaviour. We now support 51 different ACL types, and many support wildcarding behaviour.

In general, all ACL types are now case insensitive.

For those ACL types that support parts separated by delimiters (for example applications such as remote.citrix.ica where the parts are separated by "."), we support wildcarding at the delimiter level (so that for example "remote.*" would match "remote.citrix.ica"). In particular, this is true for application (application identifiers), domains (such as cloud.network-box.com), email addresses (user@domain), file content types (separated by delimiter "/"), file types (separated by space delimiters), http pathnames (separated by delimiter "/"), http hosts (similar to domains), http urls (with delimiters "/" and "." for the path and domain parts, respectively), http user agents (separated by space delimiters), ethernet mac addresses (with delimiter ":"), signature names (with delimiter "."), signature sets (with delimiter ":"), threat IDs (with delimiter ":"), and waf tags (with delimiter "/").

For IP acls, we now support subnetting. So an acl containing 10.0.0.0/8 would match 10.1.2.3.

Policy Rules in Network Box 5

In Network Box 5, the configuration language for policy rules has been standardised. We now support rule subroutines, throughout the system, and the rules language has been extended to support many rich operators. In general, a rule is expressed as:

```
<action> { <term> } [ with { <attributes> } ]
```

An <action> is the final action of the rule. If all the terms are true, then that is the result of the rule. Examples of actions are ‘permit’, ‘deny’, etc.

A <term> is usually a three part <param> <operator> <value>.
1. The <param> is an attribute of the traffic to be checked (for example, the sourceip address, sender email address, or content classification).
2. The <operator> is one of the rich operators, and usually depends on the parameter (such as =, !=, >, <, >=, <=, inacl, notinacl, contains, startswith, endswith, inrange, notinrange).
3. The <value> is the value to be compared against. Each rule can list many terms, and all terms must be true for the entire rule to match and the specified action to be the result.

The “with <attributes>” are attributes to be set should all terms of the rule be true. Examples are flags to enable quarantining, whitelisting, specifying templates for alert messages, and changing eMail message subjects.

As multiple rules can be listed for each policy, the terms within one rule act like logical AND statements, and the individual rules themselves act like OR statements. In this way, the Rules have the power of a programming language, and this power allows a multitude of different policies to be specified and enforced.

Conclusion

With apologies for the technical nature of this article, we thought it would be good to provide a clear description of the technical implementation of this important component of the Network Box 5 platform. Our new ACL/Rules engine unifies the specification and implementation of security ACLs and policy Rules throughout the Network Box 5 system. Wildcarding and case-insensitivity is now treated uniformly throughout the system, and this should help both SOC engineers and customer administrators effectively define the security and organisational policies to be enforced by Network Box appliances.
Upon Network Box’s decision to start the sunset for a specific product version an announcement will be made with the following dates:

- **Sunset Date**
  Sunset Date of a product version (hardware or software) may be announced from time-to-time by Network Box Corporation Ltd.

- **End-of-sale Date**
  End-of-sale Date of a product version (hardware or software) will be six months from the date of the sunset announcement. The product version may be available after this date, subject to any remaining stock being available.

- **End-of-support Date**
  End-of-support Date of a product version (hardware or software) will be 3 years from the date of the sunset announcement. This is the date on which Network Box will cease to provide technical support, on-site support, helpdesk support, training and spare parts. Network Box may at its own discretion provide best endeavours support.

The replacement platform NBRS-5 (Network Box 5) was released almost two years ago. The situation today is that the majority of our customer base has already migrated to Network Box 5, and all new customer deployments for the past year or so have been on that newer platform.

Our ability to address emerging security threats, and take advantage of new hardware performance and reliability features, is dramatically better in Network Box 5 than in Network Box 3. In particular, the move towards SSL secured services, and Network Box 5’s ability to enforce policy within SSL encrypted traffic, provides significant benefits for security policy enforcement.

Accordingly, we are announcing the sunset of the Network Box 3 platform as of 1st August 2015.

We released the NBRS-3 (Network Box 3) platform back in 2006, and since then it has been the core backbone for Network Box security appliances protecting thousands of our managed customer networks around the world. However, times have moved on, and the Network Box 3 platform is now almost 10 years old and hard to keep up-to-date on modern hardware.

So, what does this mean? Firstly, it does not mean we are stopping support for Network Box 3 - far from it. It simply means that today we are announcing that the Network Box 3 platform is old - new customers should not use it, and existing customers should start planning for migration to Network Box 5. In six months time (1st February 2016), we will no longer sell the Network Box 3 platform. However, we will continue to provide support and security updates for Network Box 3 for 3 more years (until 1st August 2018), but our product development focus will be on Network Box 5 and its successors.
The appliances are not based on a low power, low memory, low performance, single core CPU like most of its competitors. Instead, they use multi-core CPUs, up to 8GB RAM, up to 8GB solid state operating system storage, and up to 128GB high industrial grade flash database and log storage. Coupled with 4x 1Gbps ethernet interfaces on the S-38i (6 on the S-68i), these provide performance to handle even the most high bandwidth small or remote office connections. Full UTM+ service suites are offered with both these new appliances.

Housed in an industrial grade steel chassis, both the S-38i and S-68i can operate without moving parts - fan-less for the highest reliability.

**S-38i Hardware Specifications**

<table>
<thead>
<tr>
<th>Processor</th>
<th>64bit, 1.7GHz, 2 physical cores</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>4GB, 1333 MHz DDR3</td>
</tr>
<tr>
<td>Storage</td>
<td>1 x 64GB 2.5” SSD</td>
</tr>
<tr>
<td>Networking</td>
<td>4 x 1Gb RJ45</td>
</tr>
<tr>
<td>I/O Interface</td>
<td>1x reset button</td>
</tr>
<tr>
<td></td>
<td>1 x RJ-4S Management Console</td>
</tr>
<tr>
<td></td>
<td>2 x USB 2.0</td>
</tr>
<tr>
<td>Power Supply</td>
<td>60w (external)</td>
</tr>
</tbody>
</table>

**S-68i Hardware Specifications**

<table>
<thead>
<tr>
<th>Processor</th>
<th>64bit, 1.7GHz, 4 physical cores</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>8GB, 1333 MHz DDR3</td>
</tr>
<tr>
<td>Storage</td>
<td>1 x 128GB 2.5” SSD</td>
</tr>
<tr>
<td>Networking</td>
<td>6 x 1Gb RJ45</td>
</tr>
<tr>
<td>I/O Interface</td>
<td>1x reset button</td>
</tr>
<tr>
<td></td>
<td>1 x RJ-4S Management Console</td>
</tr>
<tr>
<td></td>
<td>2 x USB 2.0</td>
</tr>
<tr>
<td>Power Supply</td>
<td>60w (external)</td>
</tr>
</tbody>
</table>
Hackers are becoming increasingly clever at hiding their “products” in order to trick users into downloading malware. We all claim to know that, but to what extent do we really understand what’s going on?

I’ll illustrate the dedication these people have to their job by analyzing an email that was caught in our email filters on 7th July 2015:

The email was basically empty, containing just an HTTP short URL. As a user, I never open links in an email unless I know the sender, I’m expecting that email, and there is some ‘explanation’ text in the email as to why I should click on that link. Nevertheless, users less used to security are more than likely tempted to click. So, the link was http://goo.gl/DvKpsb which expanded into:


Let’s analyze this long string. The URL is actually espn.go.com, a very legitimate URL; millions click on it every day to follow their beloved sports teams. There’s nothing wrong with the URL per se. But, keep reading. After the first slash, you see “redirect?url=.....”. Now, for the sake of demonstration, copy and paste the following in your browser:


As you do that, follow what happens by looking at the status line at the bottom of your browser. You’ll see the browser go first to espn.go.com, then to www.networkboxusa.com. This is what a redirect does, what it’s supposed to do. That’s the objective of the line above – it redirects you from the main URL to somewhere else; and, really, it can be to anywhere else.
The problem is, whoever’s managing espn.go.com has created a vulnerability on their website by not properly controlling the redirect. This allows hackers to exploit the redirect, sending you anywhere the hacker wants. A redirect should not be so exposed and publicly available like that. It should be controlled so it can be used only to send you where the webmaster of the website intended to redirect you.

That said, another trick follows. The link where you’re being redirected is:


Within that link, you’re being sent to tag topa4uq55053.

Now, to the non-expert eye, this might look like a Microsoft link (www.live.com) but that’s not how domain names work. You should always read them from right to left, meaning the actual domain name here is bestoffer2015.info. This domain resolves to IP 37.1.206.32 (which is assigned to the Netherlands) and belongs to a company called 3NT Hosting Network, which is likely legitimate, and completely unaware of what’s going on with this link hosted on their server. The DNS servers for the domain are, instead, hosted in the Russian Federation.

Looking at the HTML content of the www.live.....info link, we find a java script. At this juncture, the hacker could choose to do many things but he chooses to hide another redirect into the script. In fact, the script creates random URLs, all within the domain com-1sv.net. We won’t go into the details of the java script; suffice to say it merely generates random names but the actual domain does not change - it’s always com-1sv.net.

Finally, the final page, randomly generated by the java script, in reality always points to a phishing page with images of Rachel Ray and promises of fast weight loss. We didn’t analyze what happens if you actually click on links within that page, believing that at this point the lesson is learned.

What’s the lesson, you ask?

Let’s recap – a short Google URL, pointing to espn.go.com, exploiting an error on that website to point to another link in the Netherlands, wherein a java script points to a server in Mongolia. The hacker who put all this together had to:

- Register the 2 domains in the Netherlands and Mongolia
- Hack his way into those servers or, somehow, find a way to host a redirect link on those servers
- Put the phish pages on the final server; the page looks very legitimate and well done, so time was spent to create it
- Register the short link on Google
- Create the emails, instruct the botnet to send out the spam emails to distribute the link in the hopes that someone would click

This is a lot of work that took a great deal of patience. Granted, purchasing those domains, of course, was done with a stolen credit card. Of that, we have no doubt but still, it takes a fair bit of dedication and commitment to undertake this entire process simply to cloak a link so AVs can’t catch it.
The reason why hackers are doing this now is because it's virtually impossible for any AV to follow this maze of URL links to the end, to reveal the phish or malware. If we tried following every link in every email, email delivery would come to a screeching halt. Instead, AV companies hunt for the final pages so that even if you do click on the initial link, they hope to block your browser from completing the chain of redirects. But, clearly, they can't keep up with the onslaught of chains of links. The final page, typically done by the hacker, took time to develop, and the chain of links is manipulated to change so many times that it becomes impossible to follow. And while the end link goes to the same page; that page may be hosted on several servers, and the paths left by the hacker to get you there will likely be millions, each different.

How, you might ask, how is Network Box protecting me from this threat?

In a case such as this, every aspect of the Network Box protection toolkit may come handy. As the email is received, we may be able to recognize the originating IP as blacklisted. If that is not yet the case, we may be able to recognize the URL as being blacklisted as well. Therefore, the email scanner, with its many engines, may pick up this email and quarantine it (in actuality, that is truly what happened – we picked up this email for the study from our quarantine).

It also doesn’t help that the first redirect in this case was actually linked to espn.go.com, which is legitimate. It would be nice if sites like them would check their codes and remove such vulnerabilities. Hackers already have enough tools in their arsenal. We don’t need to provide them with even more launching platforms, do we?

However, assuming none of that happens for some reason, and assuming the user clicks on the link, our web filtering and browser AV protection would analyze the content of the page and, if it contains malware, your Network Box will block the page from even loading. One caveat though; if the landing page is encrypted, this may not happen on our 3.2 platform – and this is a VERY important reason to encourage all of you, once more, to migrate to 5.0 and adopt HTTPS decoding to allow the Network Box to scan encrypted content and enhance the protection of your network.

Undeniably, the best defense against these threats is user awareness - so educate, and keep educating, your users about these threats, explain how they work, what they do, the potential dangers they pose, and make sure they don't click. Clearly, if the user does not click, the whole argument becomes moot because the threat disappears. Because, for as much as AV companies may be striving to find those "final landing pages" and create protection against them, hackers have become increasingly savvy at cloaking them, cleverly developing these long chains of redirect.

For years, we have been recommending to our clients, the idea that security spending must include a large portion dedicated to educating users to think in terms of safety. To teach them what threats may look like, ensure they understand what lurks behind a link, and establish a mindset of thinking not once, not twice, but thrice (at least) before clicking on any link. Because if they do not click, the threat cannot come in. So please, think before you click!

Stay safe.
The ONE Vulnerability

At Network Box, quite often we are asked “what should we focus on, in terms of cyber security?” Whilst there is a whole catalogue of potential threats and vulnerabilities, if there is only ONE vulnerability you should do something about before the end of this year, this is it...

The Story

Joe is the sys admin at a small textile company. He’s worked there for 15 years, and pretty much runs the place. Sure, he’s got some help for changing printer toner, updating workstation patches, etc. But, for the servers and security, he is it.

One day, Joe gets to his office for the worst day of his life. The first thing he notices is that his office desktop Apple iMac has been wiped clean. Next, his iPhone goes blank. He rushes to another workstation and finds his company website looks completely different. He doesn’t understand it – that uses HTTPS SSL and can’t have been hacked! He can’t get to his local eMail, and can’t even get into his gmail as it says his password is incorrect. His facebook wall is messed up, instagram wiped, and his twitter is a never ending stream of adult links.

Joe really can’t understand what happened. Every service he uses (Apple iCloud, his company active directory login, facebook, instagram, twitter, etc) has different usernames and passwords. How could this have happened?

The fix takes days, and both Joe and his company suffer incalculable damage to their reputations. Without an on-line way of fixing the issue, Joe has to contact each and every provider by telephone (have you ever tried to contact Instagram, Facebook, etc, by phone?) to resolve the situation. He has a horrendous time even proving who he is, in order to get back in control.

The Vulnerability

While Joe is a fictitious character, the key points of the above story are true and represent a very real threat to everyone online today. So, how did this happen, and what was the vulnerability?

Joe had correctly setup each of his on-line accounts with individual usernames and different passwords. That way, if one service was hacked (as seems to happen so often nowadays), it couldn’t affect the others. The problem is that all these services offer a ‘password recovery’ feature. The password can be reset using a secret link sent to the registered eMail address.

The eMail address Joe used was his office one. But, that is secure. It is long and complex, with mixed case letters and numbers. He changes it every few months.

Being a busy on-the-go type of guy, Joe uses his iPhone to remotely access his eMail when out of the office. When he set it up, he entered his username and password, and assumed that was secure. He also setup his outbound SMTP server to be his office server, using the same credentials for SMTP AUTH. He didn’t realize that both IMAP4 and SMTP AUTH can send those credentials in clear text, not even protected by SSL.

A couple of weeks ago, he was in an airport lounge. He connected to the free airport wifi, retrieved his eMail and sent some replies. That was his mistake.
The Attack

The attacker was also on a flight that day, and for his idea of fun turned on his laptop’s connection sharing advertised as ‘free airport wifi’, along with a data traffic logger. He recorded Joe’s IMAP4 eMail retrieval, as well as SMTP AUTH credentials for sending eMail.

Armed with that, it was trivial for the attacker to get access to Joe’s eMail account. From there he could change Joe’s facebook, instagram, etc, passwords by simply using the ‘lost password’ link. Similarly, Joe’s gmail account was compromised.

The website was compromised by DNS and SSL. Firstly, the attacker did a simple lookup on Joe’s company domain to find the DNS provider, and then could ‘lost password’ Joe’s account there to get control. By comparison, the SSL was trivial. He merely ordered a new SSL certificate online, using Joe’s eMail to verify control of Joe’s company domain. Now he could setup his own version of the website, and all Joe’s customers went to the attacker’s version of the site.

But, why wipe Joe’s iMac and iPhone? Well, just to make it harder for Joe to recover from the problem. To delay the fix. To keep the attacker in control for as long as possible. Nothing personal. Just business.

The ONE Vulnerability

So, what is that ONE vulnerability you should do something about before the end of this year? It is protection of your email accounts.

With almost all on-line services available today, your eMail address proves who you are, and the ability to view eMail you received on that address is all you need to reset online account security credentials.

It doesn’t matter if you use different passwords, on different services. It doesn’t matter if those passwords are long and complex. It all comes down to your one single password recovery eMail address.

Here are some simple steps you can take to improve your security:

• SSL secure your eMail (both SMTP and POP3/IMAP4). If you can’t do this on your mail server, Network Box 5 can do this for you. We can front an SSL certificate for you, so the remote communications to your office are secure, even if your server is not.
• Choose secure password authentication options. Make sure your server supports them, and make sure you’ve enabled them on your mail clients. Never use BASIC AUTH or other plaintext authentication mechanisms, unless the session is protected by SSL. Again, Network Box 5 can assist with this by offloading the SSL to the perimeter protection.
• Choose online services that take this seriously and offer alternative security options. For example, Google offers Authenticator for two-factor authentication.
• Consider a separation of home/office roles, and don’t cross-permit password recovery email address (especially not permitting your office services to be recovered with your personal email address).

In the above story, we’ve mentioned names and products for example only, and not to blame any one set of services. It matters little if Joe was using iPhone, Android, or Windows Phone. Pretty much all such online services suffer from these vulnerabilities, and many offer advanced protection capabilities that you can enable to help you stay secure.

Stay safe.
In 2013, Network Box 5 was launched, offering next generation security solutions to customers right across the globe. Since the launch of the platform, many new features and enhancements have been added, and Network Box is now proud to announce the launch of Network Box 5.3, the next level in cyber security.

**Network Box 5.3**
THE NEXT LEVEL IN SECURITY

### Next Generation Firewall
Network Box has taken our Hybrid Firewall which combined:
- Packet Filtering
- Stateful Inspection
- Proxying
and added Anti-DDoS as standard, with automatic bridging between IPv4 and IPv6 protocols. In addition, the Network Box IDP (Intrusion and Detection and Prevention) system can now optionally include WAF (Web Application Firewall).

### Ruggedized VPN
The Network Box VPN system has been Ruggedized to relieve connectivity issues.
- Bridges non-standard compliant connections
- Automates the approach for the most common issues
- Better handling of ISP connectivity issues
- Logging at three levels:
  - Connection start/stop events
  - Session
  - Correlation

### Visual VPN
All VPN types (IPSEC, SSL, PPTP, etc) have been unified into a single unified framework, and centralized status, reporting and control.
- Configurability of VPN types
- Reporting on VPN availability
- Capability to start/stop/restart VPN tunnels
- New Global Monitoring Sensor for VPN links
Enhanced Authentication and Encryption
Elliptical Curve Key Exchange, X.509 Certificate Authentication, and IKEv2 Support, have all been added to Network Box 5.3, so that KV-SafeNet Certification could be achieved, allowing for official integration with Germany’s medical network.

Infected LAN
Identifies infected workstations and server in the LAN/DMZ, and isolates them from the network:
- Detection of outbound access to known public BotNet command and control servers
- Detection of outbound access to known malware update sites
- Highly-granular detection for highly-prolific malware
- Optional support for dynamically blacklisting detected infected workstations / servers

Kiosk Mode
Gives an overview of users activity irrespective of the device they are using, application and IP address:
- Supports user authentication transparently - users access the network via a log in screen
- Supports iOS, Android and other mobile devices
- Able to track entity attributes and activities
- Fully integrated with holistic entities

SSL Support for encrypted traffic
Network Box also blocks malware and other undesirable content arriving via encrypted channels:
- SSL support through all our core protocols
- STARTTLS support for switching to secure mode
- SSL offload All this includes full scanning and policy enforcement

Event Correlation
The system correlates intrusion activity across all the network-frontline, network-ips, network-ids and network-firewall systems:
- Maintains statistical data for each source IP
- Highly configurable thresholds
- Dynamically blacklisting the source of the attack
- Alerts admin (via email) to the source and details of the attack

Z-Scan+
Enhanced protection with the augmentation of Z-Scan, with additional third-party threat intelligence:
- Extends the number and breadth of signatures available in the cloud
- Supplements the on the box anti-malware signature set
- Allows Network Box to be one step ahead of the attackers

Cloud email backup
Resolves loss of email issues by storing incoming emails in the cloud, and delivering them when the problem has been fixed:
- Backup queuing if the primary MX servers are overloaded and temporarily not accepting new connections/emails
- 4 geographical locations: Asia, America, Europe and the Pacific
- Once delivered, the emails will be removed and only logs will be retained
Cloud DNS backup

Allows customers to use Network Box’s extensive network of cloud DNS servers to provide backup DNS in the cloud.

- DNS queries are responded to, even if the Internet link to the master DNS server in the LAN/DMZ is down
- Full control over which of domains will use the service
- Full control over which cloud backup servers will be used for which domain

KPI reporting

The monitoring and reporting system, have been enhanced to leverage the concept of KPIs (Key performance Indicators), and is able to show what has been happening both in real-time, as well as over any given period of time.

HTML-5 Dashboard refresh

The Network Box 5.3 Dashboard has been significantly improved to include:

- Icons refreshed (clearer, self-explanatory)
- Improved workflow (spam, holistic, etc)
- KPIs and Reporting
- User-focused dashboard layout
- Single global menu
- Enhanced search function

Enhanced multi-protocol handling

Network Box 5.3 includes an enhanced Protocol Enforcement system, for better handling of non-native use of ports, by applications.

- Supports Next Generation applications such as WhatsApp and Line
- Allows or denies access for these applications
- Integrated with the Application Identification and Control system

iOS / Android App

New iOS / Android App for mobile devices will be launched.

- iOS APNS support
- Google GCM support
- Control for notifications:
  - Set-up schedules
  - Types of notifications
  - Group boxes and notifications

VPN-5

High performance VPN device for securing connections for your home office or branch office’s Virtual Private Network

- Dual Core 1.6 GHz CPU
- PPTP, SSL and IPSEC support
- High performance to handle large bandwidth remote office connections

S-38i / S-68i

UTM device to offer small offices Enterprise-level protection

- Multi-core processor
- Industrial grade chassis
- Fanless, with no moving parts

Network Box 5.3
Contents

• Cloud DNS Backup
  • Core Library Vulnerabilities
    • Proxying SSL: Part 1
    • Proxying SSL: Part 2
  • Network Box VPN-5 Model
  • VPN Reporting: Part 1
  • VPN Reporting: Part 2
• Unification of ACLs and Rules Engine
  • NBRS-3 Sunset Announcement
  • Network Box S-38i and S-68i Models
    • Network Box Special Report: Analysis of an email attack
      • The ONE Vulnerability
    • Network Box 5.3