Detect. Decrypt. Deter.

MIRA WHITEPAPER

Architecture Matters when Choosing a TLS Visibility Solution

Gaining visibility into encrypted traffic, primarily TLS, in order to detect and prevent cybersecurity threats is an important part of an Enterprise security strategy.

As TLS accounts for almost all Internet traffic these days it is commonly used by malware operators as a way to avoid detection. According to the latest Sophos Labs report, the incidence of malware using TLS has grown from 20% in 2020 to 46% in March 2021. If your security tools cannot detect and prevent threats within TLS traffic then there is a high risk of breach or compromise.

In order to protect against these types of attacks, security devices need the ability to see inside encrypted flows in order to detect and block threats. The focus of this paper is on the differing ways that security tools can gain visibility into TLS traffic flows and the strengths and weaknesses of the architectures underpinning these approaches.

Document version 1.0
October 2021

mira SECURITY

MiraSecurity.com

info@mirasecurity.com

Table of contents

Introduction

Gaining visibility into encrypted traffic, primarily TLS, in order to detect and prevent cybersecurity threats is an important part of an Enterprise security strategy. As TLS accounts for almost all Internet traffic these days it is commonly used by malware operators as a way to avoid detection. According to the latest Sophos Labs report, the incidence of malware using TLS has grown from 20% in 2020 to 46% in March 2021. If your security tools cannot detect and prevent threats within TLS traffic then there is a high risk of breach or compromise. For more details see:

https://news.sophos.com/en-us/2021/04/21/nearly-half-of-malware-now-use-tls-to-conceal-communications/

In order to protect against these types of attacks, security devices need the ability to see inside encrypted flows in order to detect and block threats. The focus of this paper is on the differing ways that security tools can gain visibility into TLS traffic flows and the strengths and weaknesses of the architectures underpinning these approaches.

Safe and secure visibility into TLS traffic

When deploying systems that provide visibility into encrypted traffic it is important to ensure that the act of gaining visibility does not compromise the end to end authentication and security of the connection. Providing "safe and secure" visibility into TLS traffic requires careful design of the systems used to provide visibility. Key requirements for safe and secure visibility are:

- The system should not downgrade the cryptographic strength of the connection. Downgrading the cryptographic strength of the connection when gaining visibility is a bad practice that weakens enterprise security.
- The system should, when possible, track and fix known TLS vulnerabilities to provide enhanced protection when either the client or server involved in the connection are susceptible to these vulnerabilities.
- The visibility system should be future proof so that the use of new TLS features does not cause the connection to fail or to be less secure. New TLS features may be as a result of development in standards groups such as the IETF or may be proprietary experimental extensions deployed by an end point provider such as a browser vendor.
- The system should provide mechanisms to allow compliance with local privacy regulations or company specific privacy requirements. Policy rules that prevent visibility into flows enable a visibility system to comply with privacy regulations.
- The system should provide the ability to guarantee data integrity even when decrypted data is passed through an in-line security tool that could modify the plaintext. If required by the customer the visibility system should be capable of ensuring that only the original data is passed on to the destination.

- The system should validate TLS certificates from servers and certificate authorities and ensure that the act of providing visibility does not hide validation status from the client or server.

A more detailed discussion of the challenges in providing safe and secure visibility can be found here:

https://docs.broadcom.com/doc/responsibly-intercepting-tls-and-the-impact-of-tls-1.3.en#:~:text=TLS%201.3%20adds%20another%20layer,vendors%20must%20both%20act%20responsibly.

Architectural differences between solutions

Before considering the different architectures used by visibility systems we first need to consider how visibility systems and security tools will be deployed in the network. Depending on the solution being used, the visibility system and the security tool may be different devices or combined in a single device.

Network deployments

While visibility systems can be deployed in-line or out of band the visibility options provided are very different between the two deployment mechanisms.

- Out of band (OOB) visibility systems work on copies of the network traffic obtained from a span or tap port. This means that the visibility system cannot participate in the TLS handshake which restricts the type of TLS traffic that can be decrypted to provide visibility. If the visibility system has a copy of the destination server's TLS certificate and private key AND the TLS handshake that establishes the session uses RSA key exchange then visibility is possible. TLS 1.3 does not support RSA key exchange so OOB inspection is not possible for TLS 1.3 traffic. OOB inspection can also not be used to inspect traffic to servers outside the Enterprise as it is impossible to obtain the private key for those servers.

  Recently there have been some OOB visibility systems developed that overcome the above restrictions by providing the system with a copy of the ephemeral key negotiated during the TLS handshake which can be used to decrypt the copy of the flow. Such systems require proprietary software on the server that discovers and exports the ephemeral key to the OOB visibility system and are typically focused on inspecting inbound TLS traffic destined for a server within the Enterprise. In theory such systems could also use an ephemeral key exported from the client rather than the server but deploying proprietary software on all the clients within an Enterprise creates management and scaling issues. Ensuring security for the ephemeral key when it is sent to the OOB visibility system is a major issue as an attacker who could collect the ephemeral keys could use them to decrypt captured encrypted flows after the fact as well as in real time.

  As OOB systems work on a copy of the traffic many of the issues regarding "safe and secure" visibility do not apply. OOB systems can be used to feed passive security tools only as the decrypted flow received by the tool is a copy of the live network flow so an active tool that modified this flow would not have any effect. So, feeding an IDS makes sense from an OOB visibility system whereas feeding an IPS does not.

- In-band visibility systems work on the live traffic stream and can actively intervene in the TLS handshake to insert themselves as a controlled Man In The Middle (MITM) device. Such systems can provide visibility into all TLS traffic, including TLS 1.3 traffic and can be used when either the server or the client is under control of the Enterprise. As the visibility system participates in the TLS handshake it can influence the cryptographic strength of the end to end connection so all of the issue around safe and secure visibility apply to these systems

  As in-band visibility systems are working on the live traffic stream they can be used to feed both passive and active security tools. In the case of in-line tools that may modify the decrypted data an in-band system can provide data integrity control to prevent such modifications being propagated to the destination, if required by the Enterprise.

  It is possible to have an in-line visibility system that is receiving ephemeral keys and using these to decrypt the live traffic flow. Such a system would not participate in the TLS handshake but would allow an active security tool to modify the decrypted traffic. In this type of deployment the ephemeral keys need to be received by the visibility system before the encrypted payload packets arrive to avoid creating additional latency and a requirement for buffering. As with OOB ephemeral key systems the main use case is inspecting traffic to servers within the Enterprise for the same reasons as noted earlier.

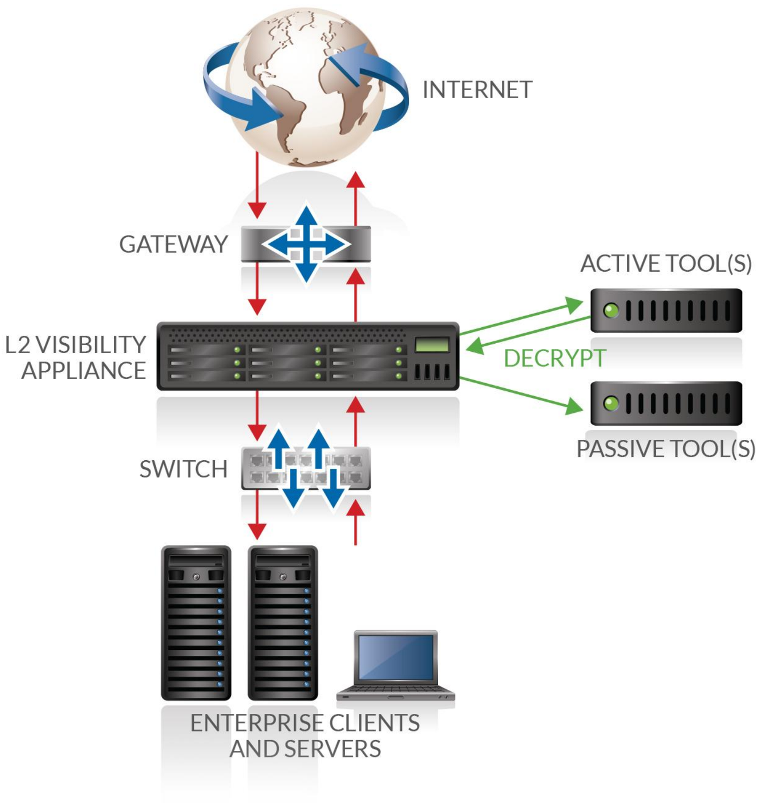
Different Architectural Approaches

The following section looks at various approaches taken by different visibility products and security products with inbuilt visibility capabilities.

- Transparent visibility systems that are focused solely on providing visibility into TLS and then feeding attached passive and active security tools. Such systems act as transparent Layer 2 devices forming a “bump in the wire” or “bump in the tunnel” from a network perspective.
- Application Delivery Controllers (ADC) can be used to address TLS visibility by terminating and re-originating TLS sessions with a zone of decryption between the devices in which the security tools are located. Such systems act as Layer 3 devices terminating links and tunnels on one side of the zone of decryption and re-generating them on the other side.
- Next Generation Firewalls (NGFW) can address TLS visibility by terminating and re-originating TLS sessions within the device and feeding decrypted data to security tools that are part of the NGFW or are attached to it.

Transparent Visibility Systems

TRANSPARENT VISIBILITY DEVICE



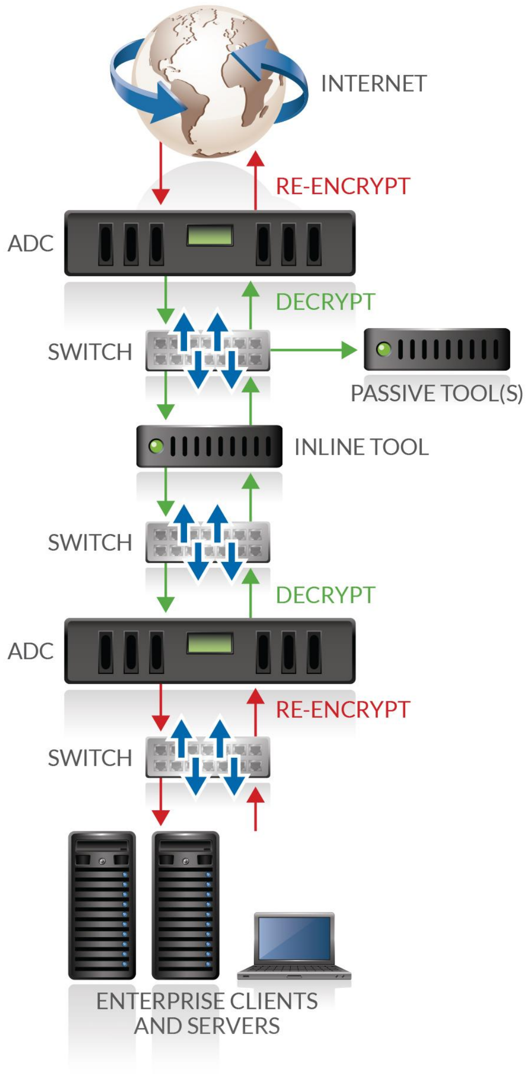
The figure above shows a typical L2 transparent visibility solution with the system being a bump in the wire and with both inline and passive security tools being fed decrypted traffic. In most cases the security tools will also be fed all the other flows passing through the bump in the wire. This allows the tools to see non TLS flows and TLS flows that are not decrypted due to policy and to perform whatever security checks are possible on these flows. As there are no IP addresses associated with the bump in the wire the attached tools are directly connected ensuring that the decrypted flows are not sent over a network where they could be intercepted.

Most systems will automatically detect TLS traffic no matter what destination port is being used by the TCP connection which simplifies configuration and policy definition. Also, such systems do not care what protocol is running on top of TLS as they simply decrypt the flow and hand it to the security tools. This allows visibility into any protocol, not just HTTPS. In networks that use VLANs or tunnels such as GRE or VXLAN to create virtual overlay networks a system that can act as a “bump in the tunnel” detecting TLS and providing visibility for TLS flows within the tunnel without requiring the tunnel to be terminated and re-originated greatly simplifies deployment. A good L2 transparent visibility system should not require any changes to the existing network addressing or routing.

As the decryption and re-encryption necessary to provide visibility into TLS are carried out in the same device it is possible to implement data integrity features that ensure that modification of the decrypted flow by inline security devices does not change the data sent between the client and server.

ADC Visibility Systems

APPLICATION DELIVERY CONTROLLER



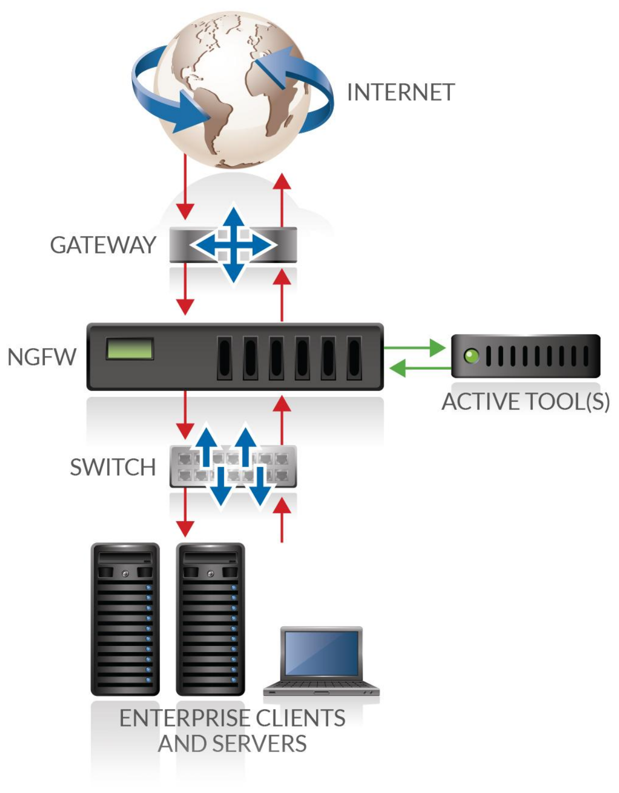
The figure above shows a typical ADC visibility solution with two ADC devices on either side of an uncontrolled zone of decryption that the security tools are located within. The ADC interfaces have their own IP addresses so it is necessary to change the existing network addressing and routing in order to deploy the visibility solution. Tools within the uncontrolled zone of decryption are typically attached to a subnet that exists between the two ADC ports so it is possible with access to this subnet to intercept or snoop on the decrypted traffic. ADC solutions support directing traffic other than decrypted TLS traffic to the security tools if required.

The ADC will often use the destination port number as the way to determine which flows are TLS, for example looking for port 443. Normally it is possible to configure things so that additional ports are treated as being TLS, but this increases the complexity of configuring the solution. Internally the ADC often implements a proxy to handle the decryption and this may limit the types of protocol running over TLS that can be handled. Some ADC solutions have only recently added the ability to provide visibility into HTTP/2 traffic relying on the ability to downgrade the flow to HTTP/1. In a network that uses VLANs or tunnels to create overlay networks deploying an ADC visibility solution is complex as all the VLANs and tunnels need to be terminated by the first ADC on ingress and re-originated by the second ADC on egress. In addition new tunnels may need to be created within the zone of uncontrolled decryption if tunnel details are used as part of the security policy in the security tools.

As the decryption and re-encryption necessary to provide visibility into TLS are carried out by different ADCs it is not possible to implement data integrity features.

NGFW Visibility Systems

NEXT-GENERATION FIREWALL



The figure above shows a typical NGFW visibility solution with decrypted traffic being made available to internal security applications as well as to externally connected security tools. As the same resources are being used to implement the NGFW functions and its internal security applications, enabling visibility into TLS traffic has a significant impact on performance as TLS decryption and re-encryption requires considerable resources which are no longer available to the rest of the NGFW. Most NGFW solutions are optimized for sharing decrypted data with internal security tools but some also have the ability to send this traffic to external security tools as well via an external interface. One limitation in feeding external tools in some cases is that only decrypted traffic can be sent over the interface preventing the tool from seeing other traffic at the same time.

Similar to the ADC, the NGFW will often use the destination port number as the way to determine which flows are TLS, for example looking for port 443. It is normally possible to configure that additional ports are treated as being TLS but this increases the complexity of configuring the solution. Internally the NGFW often implements a proxy to handle the decryption and this may limit the types of protocol running over TLS that can be handled. Some NGFW solutions have only recently added the ability to provide visibility into HTTP/2 traffic relying on the ability to downgrade the flow to HTTP/1.

As the decryption and re-encryption necessary to provide visibility into TLS are carried out by a proxy within the NGFW there is typically no mechanism to provide data integrity to prevent modifications to the data while decrypted being sent to the destination..

Impact of different architectures on TLS visibility

The key differences between the three approaches detailed above are summarized in the matrix below.

Feature	Transparent	ADC	NGFW
Network transparent at Layer 2	✓	✗	✗
Controlled decryption zone allowing data integrity	✓	✗	✗
Performance impact on other features	✗	✓	✓
Ease of deployment	✓	✓	✗
Automatic detection of TLS flow - port / protocol agnostic	✓	✗	✗
Supports and protocol running over TLS	✓	✗	✗
Comprehensive support for existing and upcoming TLS features	✓	✗	✗
No impact on existing network topology, addressing and routing	✓	✗	✗

The issues that matter to a specific Enterprise will vary depending on the type of business and its security profile , the existing network topology before a visibility solution is deployed, regulatory regimes affecting the Enterprise etc.

A few examples of when particular issues are important are provided below.

- In financial services or banking it may be essential to guarantee that even though decrypted traffic passes through one or more inline tool it can never be modified. The Enterprise needs to guarantee that the data sent by the client is what the server receives and vice versa. Relying on the configuration of inline tools to prevent any modification of the data runs the risk of accidental or malicious misconfiguration resulting in changes to the data while it is decrypted. If the visibility solution in use does not provide a data integrity option then these changes will be re-encrypted and sent on to the destination breaking the guarantee required by the Enterprise.
- In a large data center environment where VXLAN tunnels are used to create overlay networks to simplify management of different types of traffic, deploying a visibility solution that does not support a transparent "bump in the tunnel" approach will require significant work to maintain the overlay networks when a visibility solution is deployed.
- In a network with an existing NGFW solution that is not providing visibility into TLS the simple solution may be to turn on this feature when it is decided that visibility is essential for security. However, the resulting performance impact on the NGFW may mean that it no longer has the capacity to handle the traffic load from the network. One option is to upgrade to a higher capacity NGFW with sufficient additional resources to handle the network load while providing visibility. Another approach is to deploy a transparent visibility solution with the NGFW attached as an inline tool. This allows the existing NGFW to only have to deal with decrypted traffic so it will continue to have the capacity to handle the network load. The relative costs of the two options then become the determining factor in which solution to deploy.
- An Enterprise may want to avoid in-line devices but still want to have visibility into both inbound and outbound TLS traffic to feed passive security tools. All of the solutions discussed in earlier sections require the visibility device to be inline. The only solution that could meet the requirement is a system that received ephemeral keys exported from servers within the Enterprise or from client systems within the Enterprise. Deploying the visibility device is the simple part, the complexity in this type of solution is in deploying software on servers and clients that can acquire the ephemeral keys and export them to the visibility device in a timely fashion. If agent software is available for the Enterprise servers and client systems, then there are management and compliance issues that need to be overcome before it can be used. Will it be acceptable to install the agent software on the server providing Internet banking to customers? If so, how long is the qualification and testing period before it is certified for use?

Future proofing TLS visibility solutions

TLS is not a static standard, there is ongoing work to develop and enhance the features it supports both within the IETF TLS working group and by large service operators and client software providers. Visibility solutions need to be designed with the expectation that new features will have to be dealt with.

For example, there is currently activity in the IETF to define support for "Delegated Credentials." This work allows a server to provide frequent updates to the private/public key pair that are used during the TLS handshake. Normally this key pair is part of the server certificate issued by a public CA and only changes when the certificate is updated. By delegating authority to the server to allow it to create a new key pair, security is increased as the key pair can be rotated more frequently, in hours or days rather than months or years. This work is not yet (Sept 2021) a released RFC but Firefox has an implementation in their browser and CloudFlare and Facebook have implementations on their servers.

This means that a visibility appliance is already likely to see TLS client hello messages that contain an extension indicating that the client supports Delegated Credentials. If this is passed through to Facebook or CloudFlare then the server will use delegated credentials in the handshake. However, if the visibility device does not yet implement support for Delegated Credentials and so uses the key pair in the server certificate the handshake will fail and the session will not be established.

The solution to this in the long term is for the visibility appliance to support Delegated Credentials but in the short term it can prevent their use by removing the extension in the Client Hello message that indicates the client supports this feature. This will cause the server not to use Delegated Credentials and to rely on the key pair in the server certificate.

A "safe and secure" visibility solution will prevent problems such as this by ensuring that only extensions for features that it recognizes and can support are allowed to pass from the client to the server. Any unrecognized extensions or extensions that are not currently supported will be removed to prevent the server from using these features. Taking a proactive approach ensures that new TLS features don't break existing visibility systems and cause disruption to user traffic.

MiraSecurity.com

info@mirasecurity.com

Mira Security – US Headquarters
330 Perry Highway, Suite A
Harmony, PA 16037
Phone: +1 (412) 533-7830

Email: info@mirasecurity.com
mirasecurity.com