Security and Privacy on the Encrypted Network

Version 1.5
Released: January 20, 2015

Securosis, L.L.C. 515 E. Carefree Highway Suite #766 Phoenix, AZ 85085 **T** 602-412-3051
info@securosis.com www.securosis.com

Author's Note

The content in this report was developed independently of any sponsors. It is based on material originally posted on the Securosis blog, but has been enhanced, reviewed, and professionally edited.

Special thanks to Chris Pepper for editing and content support.

Copyright

Security and Privacy on the Encrypted Network

Table of Contents

The Future is Encrypted

The cloud and mobility are disrupting the way IT builds and delivers value for organizations. Perhaps you are moving computing workloads to the cloud and storing an increasing amount of data outside the corporate perimeter, or maybe an increasing proportion of employees now access data from outside your corporate network, but you no longer have overarching control over networks or devices. So security teams need to adapt their models to protect data. For details of this disruption see our recent Future of Security research.

> At some point during every attack the adversary becomes an insider with credentials to access your most sensitive stuff.

But that isn't the only reason organizations are finding themselves forced to adapt security postures. The often-discussed yet infrequently addressed 'insider threat' can no longer be ignored. Given how attackers are compromising devices, performing reconnaissance to find vulnerable targets, and sniffing network traffic to steal credentials, at some point during every attack the adversary becomes an insider with credentials to access your most sensitive stuff. Whether an adversary is external or internal, at some point they will get *inside* your network.

Finally, tighter collaboration between business partners means people outside your organization need access to your systems and *vice-versa*. This access should not impose substantial additional risk to your environment, so those connections need to be protected to ensure data is not stolen.

These trends mean organizations have no choice but to encrypt more traffic on their networks. Encrypting the network prevents adversaries from sniffing traffic to steal credentials and ensures data moving outside the organization is protected from man-in-the-middle attacks. So we expect to see a much greater percentage of both internal and external network traffic to be encrypted over the next 2-3 years.

> Network encryption impacts traffic inspection and enforcement of security policies. Encrypted networks also complicate security monitoring because traffic needs to be decrypted at wire speed for capture and forensics.

Why Decrypt?

But no good deed goes unpunished. Network encryption adversely impacts your ability to inspect traffic and enforce security policies. Thus the increase in encrypted traffic impacts your visibility to detect sensitive content leaving your network. This data loss isn't always related to advanced attackers exfiltrating your data, as employees may inadvertently send

information outside the network within an encrypted session. Encrypted networks also complicate security monitoring because traffic needs to be decrypted at wire speed for capture and forensics. Unless you have a strategy for encrypted traffic management, you run the risk of missing critical data leakage.

But you'll need to consider the other implications to decrypting network traffic. For example, decrypting traffic also presents compliance issues and raises human resources considerations, which must be factored into your plans as you grapple with increasing encrypted traffic, appearing deeper into your networks.

This paper will discuss security policies to ensure that data doesn't leak out over encrypted tunnels, and that employees adhere to corporate acceptable use policies, by decrypting traffic as needed. Then we will dive into security monitoring and forensics, discussing traffic decryption strategies to ensure you can properly alert on security events and investigate incidents. We will wrap up with guidance on how to handle human resources and compliance issues, as an increasing fraction of network traffic is encrypted.

Use Cases

As we mentioned above, our inability to inspect encrypted traffic impairs our ability to enforce security controls/policies and meet compliance mandates. We will dig into how to strategically decrypt traffic to address key use cases, including security policy enforcement and monitoring for security and compliance. We also need to factor in the HR and privacy issues associated with decrypting traffic — you don't want to end up on the wrong side of a worker council protesting your network security approach.

What to Decrypt?

The first step in gaining visibility into the encrypted network is to set policies for when traffic will be decrypted and how long it will remain decrypted, as data remains unprotected (in decrypted form) until you re-encrypt it . These decisions are driven by organizational culture so you need to figure out what will work for your organization. As security guys we favor more decryption than less, to enable more comprehensive inspection, and therefore stronger monitoring and enforcement. But this is a company-specific choice.

As security guys we favor more decryption than less, to enable more comprehensive inspection, and therefore stronger monitoring and enforcement. But this is a company-specific choice.

Several factors influence decryption policies, most obviously the applications themselves. You are most likely to decrypt a few main applications:

1. **Webmail:** Employees think they are doing your organization a favor by working at all hours of the day. But this always-on workforce uses personal devices and may decide (however unwisely) that it is easiest to send work documents to personal machines via personal email accounts. What could possibly go wrong? And of course there are plenty of malicious uses for webmail in a corporate environment with it being a common means of exfiltrating sensitive data. There are endpoint DLP agents to catch this behavior, but if you don't have them deployed you should inspect outbound webmail traffic and look for keywords or content that shouldn't be sent outside of the organization. As most webmail is now encrypted, you need to decrypt these sessions before you can inspect the traffic.

2. **Web browsing:** Social media sites and other web properties use user-generated content that may be protected or sensitive to your organization, so you need to ensure you can enforce policies on general web application traffic as well. Applications increasingly use SSL/TLS by default, so you will need to decrypt to enforce acceptable use policies and protect data.

3. **SaaS Applications:** Business functions are increasingly migrating to Software as a Service (SaaS) so you need to inspect SaaS traffic. You may want to enforce tighter content policies on SaaS apps because you don't control the security of the data in a SaaS environment, but first you need to decrypt the traffic for inspection and enforcement. Also be aware that some of these apps may use non-standard ports (besides the typical TCP 443), which makes it more complicated to inspect the traffic of that specific application.

4. **Custom Applications:** Custom web applications and partner web applications also require scrutiny. Given the likelihood they will use sensitive data. As with SaaS applications you will want to enforce granular policies for these applications, so you need to decrypt.

To net it out, if an application has access to protected or critical data you should decrypt and inspect its traffic. But secondary attributes may preclude or demand decryption. For example certain web apps/sites, such as consumer healthcare and financial sites, should be whitelisted (never decrypted) because they handle private employee data.

Another policy trigger will be individual employees and groups. Maybe you don't want to decrypt traffic from the legal team because it is likely protected and sensitive. And of course some folks require exceptions. Like the CEO, who gets to do whatever he/she wants and may approve an exception for her/his own traffic. There will be other exceptions — we guarantee it — so make sure your policies are flexible enough to support your business requirements and culture. For example one app might need to always be inspected for *all* users due to the sensitivity of its data. Likewise perhaps one set of users won't have their traffic inspected at all because of the sensitivity of that job. You should have flexibility to decrypt traffic to enforce policies, based on applications and users/groups, to precisely map to business processes and requirements.

Where to Decrypt?

Now that you know *what* to decrypt, you need to determine the best place to do it. The answer hinges on which applications need to be inspected and which devices need the data for monitoring and/or enforcement.

1. **Firewall:** Firewalls frequently take on the decryption role because they are inline for both egress and ingress and already enforce policies — especially as they evolve toward application-aware Next Generation Firewalls (NGFW). Decryption is very computationally demanding so you may face scaling issues, even for larger and more powerful firewalls.

2. **IPS:** As an inspection technology IPS is ineffective if it cannot inspect encrypted traffic. To address this some organizations decrypt on their IPS devices. The IPS function is more computationally demanding than enforcing access control policies on a firewall, so dedicated IPS devices tend to have more horsepower, which helps with decryption. But scalability can still be an issue under heavy decryption load.

3. **Secure web gateway:** As inspection and enforcement devices for web traffic, web filters need to decrypt traffic as well. They tend to be a bit underpowered compared to other devices in the perimeter, so unless there is minimal encrypted traffic they can run out of gas quickly and bog down outbound web traffic.

4. **Dedicated SSL decryption device**: For organizations with a lot of encrypted traffic, which is increasingly common, dedicated decryption devices which specialize in decrypting traffic without disrupting employees are available. They offer flexibility in how to route decrypted traffic for active controls (FW, IPS, web filter, etc.) and monitoring, and then re-encrypting traffic which needs to continue out to the Internet. We will get into specifics of selecting and deploying these devices below.

5. **Cloud-based offerings:** As Security as a Service (SECaaS) offerings mature, organizations have the option to decrypt in the cloud, offloading responsibility for scalability to a service provider. But this requires potentially sensitive data to be decrypted and inspected in the cloud, which may require cultural or regulatory challenge.

All these devices are typically deployed inside your network perimeter, so you remain blind to attackers encrypting internal reconnaissance traffic, or traffic moving out of the data center to a staging server inside your internal network. To address these issues you might choose to configure an existing device (most likely a firewall or IPS) internally in front of your data center, to decrypt, enforce security policies, and inspect traffic. We increasingly see this deployment model for network security gear, although internal network traffic characteristics vary from perimeter traffic. Scalability is crucial in data center environments, which typically have multi-gigabit internal networks.

To ensure scalability you will want to test the performance impact of decryption. According to independent lab tests by organizations such as NSS Labs, you may see a performance penalty of up to 80% when decrypting on firewalls and IPS devices. Obviously adequate throughput for typical traffic volumes is a fundamental architectural and deployment requirement.

> It is very common for sophisticated attackers to encrypt traffic to and from compromised devices. If you don't decrypt both ingress and egress traffic you are blind to certain attacks.

Enforcing Security Policies

Our first use case is active enforcement of security policies. This involves decrypting traffic and then enforcing policies using traditional devices — including

firewalls, IPS, web filters, load balancers, etc. It is very common for sophisticated attackers to encrypt traffic to and from compromised devices. If you don't decrypt both ingress and egress traffic you are blind to certain attacks. You can miss newly compromised machines connecting to the mothership (bot controller), along with the resulting malware downloads, because C&C traffic is encrypted. Another blind spot is exfiltration of sensitive data, which is consistently encrypted.

The table below breaks out a few of the devices where you need decrypted traffic to properly enforce the policies.

	Why does this device need decrypted traffic?	**Native decryption capabilities**
IPS	Application and network data required for analysis is obscured by encryption; cannot match to intrusion signatures	Limited. Significant performance hit.
NGFW	Application classification typically requires deep packet inspection that is encrypted; Integrated threat protection (IPS) need decrypted traffic (see above)	SSL decryption typically degrades performance by 50-80%.
DLP	DLP requires inspection of data payload to match against controlled data (e.g. credit card numbers, SSN, etc).	None
Network-based malware sandbox	Files carrying malicious payload often travel via encrypted tunnels (SSL). Without decryption malware files are not detonated for analysis.	None
Web Gateway	Web gateways need to decrypt HTTP requests to check against filtering policies; Integration with anti-malware gateways and network-based sandboxes need to decrypt those files.	Proxy-level web gateways can decrypt SSL. Potentially significant degradation of device performance.

The key considerations for decrypting traffic and sending it to an active security device are:

1. **Throughput:** Networks continue to get faster, and you need to inspect traffic at (or very close to) wire speed. Decryption and re-encryption are very resource intensive, so you need to make sure your decryption capability can scale sufficiently on the device you choose for decryption. Assume that (for now anyway) 20-40% of your traffic will be encrypted and plan for decryption throughput accordingly.

2. **Latency:** Many applications are real-time or highly interactive, so you cannot afford to introduce major latency. You need to make sure that along with adequate throughput, scaling up doesn't add unacceptable latency.

3. **Full protocol support:** Attackers can hide encrypted traffic in other protocols on different ports, so it is important for inspection engines to analyze the full traffic stream — not just port 443.

4. **Policy granularity:** Precise policies are necessary to control what gets decrypted by attributes such as protocol, user/group, application, and web site category, in order to maintain the privacy of users.

5. **Send decrypted traffic to multiple devices:** You may want both an IPS and a network-based malware sandbox to analyze traffic, so you should have the ability to send it to multiple devices without fuss.

6. **Fail open or closed:** If decryption fails will you just allow traffic to pass through, or will you block it? This can get sticky — get sign-offs from both the senior management and legal teams. Continuing business operation is often prioritized over the possibility of an attack.

Given the performance impact of decryption you will need to manage expectations all along the way. You likely cannot buy enough decryption gear to decrypt *all* traffic without any performance impact. It is very common for sophisticated attackers to encrypt traffic to and from compromised devices. So for traffic that needs to be decrypted and inspected make sure everyone understands the potential impact. A little proactive communication with key stakeholders is essential for success.

Given the performance impact of decryption you will need to manage expectations all along the way. You likely cannot buy enough decryption gear to decrypt all traffic without any performance impact.

Monitoring and Forensics

Improving incident response in the face of ever-more-sophisticated attacks requires that you monitor and capture more traffic for alerting and forensics. This use case differs significantly from enforcing security policies because you aren't quickly re-encrypting or discarding data. The goal is to either derive metadata from the packet stream or actually capture packets for subsequent investigation.

When you decrypt to enforce a security policy the data may be unencrypted for a few seconds. But when you decrypt for monitoring and forensics the data may remain unencrypted indefinitely. You need to be sensitive to this change, and far more careful and stringent about how and how long you keep unencrypted data.

When you decrypt for monitoring and forensics the data may remain unencrypted indefinitely. You need to be sensitive to this change, and far more careful and stringent about how and how long you keep unencrypted data.

Earlier we talked about kinds of applications and other attributes that might trigger or prevent decryption; use the same approach to define policies for monitoring and forensics. You also need a risk analysis on pretty much every policy, determining whether its traffic could contain sensitive information (either corporate or personal) and the risk of capturing it.

For example you might want to decrypt traffic from HR and send it to the IPS to check for attacks because HR is a frequent phishing target. But you might avoid sending that decrypted stream to your packet capture device because sensitive personnel information could be captured, posing an unacceptable risk. There are no universal right or wrong answers about what to decrypt or not — you need to ask the right questions when setting up policies.

This situation is non-optimal for security — it doesn't fit well with our general approach of capturing as much as we can and keeping it as long as we can afford. Unfortunately privacy issues, described in more detail below, demand tough choices about what you can decrypt for monitoring and how long you can keep it. But even in places where packet capture is a no-go, you will still want to decrypt and analyze session metadata (source, destination, protocol, amount of data, application, etc.) to glean patterns for analysis by your SIEM or other security analysis capability. The caveat is that in some geographies you cannot even do that basic traffic inspection, and adherence to local law generally trumps corporate policy.

You may be able to allay fears about capturing encrypted traffic by highlighting the security of the capture environment. If captured data is stored in a purpose-built device with proper access control and authorization protections, and the data is protected at rest somehow… that may reassure key influencers about decryption policies.

The table below lists the monitoring devices that need decrypted traffic:

	Why does this device need decrypted traffic?	**Native decryption capabilities**
Network Forensics/Security Analytics	Capture, indexing and searching of packet payloads require decrypted traffic.	None.
SIEM	Analysis of applications, protocols, amount of data and other session metadata requires decrypted network stream.	None.

HR and Compliance Issues

As we alluded to earlier, implementing decryption policies raise non-trivial privacy and compliance ramifications, and at some point — earlier rather than later — lawyers should review policies. Here are a few issues to consider:

1. **Geographic variations:** If you do business in a very privacy-sensitive geography you should get the local team engaged as soon as possible — especially in Europe. You will likely need to work with local authorities and/or worker councils to make sure decryption policies don't violate local laws (such as PCI-DSS, PIPEDA, FISMA, Australian Privacy Act, Germany's Data Protection Act, etc.) and compliance mandates (in addition to regulatory/legal mandates). Some locales specifically restrict website impersonation, including man-in-the-middle decryption of traffic using an intermediate certificate. There is also considerable sensitivity about employee surveillance, so consider local attitudes there as well.

2. **Whitelisting:** To address some of these issues you should consider whitelisting certain categories of applications and websites — typically healthcare and financial companies. This ensures you do not decrypt sites where the likelihood is high of intercepting sensitive or protected information. Of course this is well known to attackers, who may use compromised servers in whitelisted categories to evade decryption.

3. **Culture:** Some organizations are more sensitive to protecting intellectual property and critical data, and push for stronger monitoring and security. If you are in such an environment you may be able to decrypt more. Others are very conservative and will not monitor if there is even a slight chance employees could feel alienated or violated. This is why working with senior management and legal counsel is so important when defining policies. Avoid making a unilateral call on what to decrypt and what to bypass.

4. **Documentation:** Even if everyone signs off on policies, they may have issues later on. So make sure you can substantiate exactly what gets decrypted and when through solid documentation. Also make sure you can prove who has access to the data (especially when keeping it for monitoring and forensics), and document the investigation workflows to show who accessed what data during an investigation. Do this by leveraging logs from decryption gear, and keep reports of policies and response process workflows to prove what you are doing — and what you aren't. Better have too much documentation than ask regulatory bodies like worker councils (prevalent in EMEA) to blindly trust you.

As usual, what should be a fairly straightforward technical discussion of what to decrypt and how devolves into the squishy reality of policy and privacy mandates. That is the game we have to play as security folks. Build your policies using an inclusive process to develop and update what gets deployed according to what is acceptable in your organization.

Selection Criteria and Deployment

You now understand how to set policies for decryption, the specific use cases driving decryption of network traffic, and the human resources and compliance considerations for building policies. Next you need to understand the technical nuances of actually figuring out where to decrypt, and how to select and deploy the technology. First let's talk a bit about whether you need a standalone device.

Standalone or Integrated?

As discussed earlier, you have many choices for where to decrypt traffic, including firewalls (both NGFW and UTM), IPS, load balancers, and web & email security gateways. Obviously using an existing device for both decryption and inspection is simplest. You don't have to add other boxes or risk messing up your network addressing scheme, and you can enforce policies right at the point of decryption/inspection. A security device can decrypt traffic, inspect it, apply policy, and re-encrypt — all within a single box. For environments with limited network volumes and simple policies, integrated devices are an excellent choice.

But organizations which need to decrypt substantial network traffic tend to quickly crush the performance of existing security devices if they try to decrypt on them; onboard decryption may reduce performance of other security devices by up to 80%.

But organizations which need to decrypt substantial network traffic tend to quickly crush the performance of existing security devices if they try to decrypt on them; onboard decryption may reduce performance of other security devices by up to 80%. If you have plenty of performance headroom on existing devices you can afford the overhead of decryption. If you don't you will need to look at another device to offload decryption load, in order to let your security devices do what they do best: inspect traffic, apply policies, and monitor or capture traffic.

If you deploy various cloud, mobile and web applications that require more complicated (or advanced) policies, such as having to send traffic to multiple active and/or monitoring devices and have multiple policy triggers across the entire network stream, and cannot limit inspection to only port 443 (HTTPS) traffic an integrated device's relatively simple decryption policies may be insufficient. Additionally, if you need to send decrypted traffic multiple places, such as a SIEM or network packet capture/forensics device, an integrated solution may not be a good fit.

Also consider the cost of an integrated solution versus a dedicated solution option over time. If you have a large number of devices requiring decrypted traffic (as described above) and/or you expect an increasing amount of encrypted traffic on your network, the need to increase the size and capacity of the native devices to scale appropriately may be cost prohibitive. The cost of additional CPU, memory or networking connectivity on the integrated devices can potentially double our triple the cost of that solution. Don't forget the operational costs as well, as managing cryptographic keys and certificates across multiple native devices can also add operational cost and complexity to the integrated option.

We like integrated options when they are viable but pragmatism demands the right tool for the job, and adding decryption to another device is rarely suitable for high traffic environments or environments needed to send traffic to multiple devices. If onboard decryption can meet your performance and policy requirements, use it.

Selection Criteria

If you decide to use a dedicated decryption device, what should you look for? Here are a few consideration:

1. **Performance:** Much of the value of dedicated hardware lies in its ability scale up with traffic volume. Make sure any device or devices you buy can scale to current and future bandwidth volumes. Don't paint yourself into a corner by buying equipment that will need to be replaced as traffic volume grows.

2. **All Port Support:** One of the easiest evasion techniques for attackers is to simply change the port number of outbound traffic, sending encrypted traffic where it is not expected or monitored. Inspection devices cannot afford to trust port numbers — you need deep packet inspection to detect evasion.

3. **Accuracy:** Decryption strategy is highly policy dependent, so success requires accurate identification and categorization of traffic. Along with examining the full traffic stream, you need to ensure your devices accurately find encrypted traffic and categorize it effectively.

4. **Policy actions:** Make sure your device supports a variety of different actions on a policy hit. You should be able to decrypt, not decrypt, drop the session, or reject the session (with an HTTP error code). You also want the ability to list sources or destinations to always decrypt (blacklist) or never decrypt (whitelist), by group or user.

5. **Website category/reputation support:** We spent a long time on the nuances of setting policies — which typically consider websites, IP addresses, and applications. Given how quickly website reputations and categories change (minutes, if not seconds), it is important to have a dynamic source of current information to base policies on. That usually means some kind of cloud-based website categorization service for whitelisting (financial and healthcare) or blacklisting (known malicious IP address), along with dynamic reputation scoring for websites and applications.

6. **Multiple device support:** Given the varied decryption use cases, these devices should be flexible in how they forward traffic, and to how many devices. For example you might want to send traffic to an IPS for active control, and also to a SIEM or packet capture device for monitoring and forensics. It is important for decryption devices to interoperate natively with security devices, so that (for instance) when an IPS detects (decrypted) attack traffic, the decryption device can drop that session without human intervention.

7. **Security:** These are security devices, so you need to ensure that decryption/resigning keys and data on devices are protected. You also want the ability to reject/drop sessions if their security is insufficient. For example a weak encryption cipher could put data at risk; or it might violate policy to transmit encrypted data which cannot be decrypted by the security device, in order to prevent unknown data from leaving your environment.

8. **Transparency:** It is also important to ensure decryption doesn't impact application behavior or user interaction. End users should not need to concern themselves with security inspection. Further, the decryption device shouldn't alter packet headers in any way, as that might impair other security devices' inspection. Of course, you'll (probably) need to inform employees that they can be monitored (via the Internet use agreement), but the decryption device shouldn't impact user experience.

9. **Deployment flexibility:** There are a variety of use cases for decryption, so you want a device that supports multiple deployment models to support your various present and future requirements to decrypt traffic. For devices with multiple ports you should have flexibility in how to assign ports, as well as which ports will be active or passive, as discussed below.

Deployment

Decryption device deployment should be as non-disruptive as possible. You don't want to mess around with IP addressing schemes, force every user to see (or click) a security warning every time they make an SSL connection, or have the device manipulate IP address headers and screw up traffic monitoring and analysis. You want transparency.

Make sure you see all relevant traffic. Don't assume that all encrypted traffic will be on certain ports. Attackers frequently hide encrypted traffic on odd ports to evade decryption. So examine all traffic for encrypted data — not just the obvious ones (HTTPS, FTPS, SMTPS, POP3S, etc.) to make sure you don't miss anything.

Don't assume that all encrypted traffic will be on certain ports. Attackers frequently hide encrypted traffic on odd ports to evade decryption.

There are a few deployment options if you choose a dedicated decryption device:

1. **Passive Inline:** In this configuration the decryption device is positioned as a bump in the wire — inline — to ensure that all traffic can be inspected and decrypted according to policy. Once traffic is decrypted the device can forward it to a variety of different security devices for inspection/monitoring, load balancing between them if throughput is an issue. In this passive deployment the decryption device cannot drop or block sessions itself, nor can it re-encrypt traffic — it just copies traffic to inspection devices, forwarding the original encrypted traffic on to the network.

2. **Active Inline:** This configuration is similar to passive inline, but in this mode the device *can* enforce policies. Based on policy the device decrypts traffic and forwards it for inspection. It queues each encrypted session until it receives a verdict back from another security device. If the session is approved it re-encrypts and sends the traffic on its way. Sessions that look like an attack are dropped.

3. **Passive Tap:** In this mode the decryption device connects to a network tap or span port on a switch to receive a copy of all traffic. The decryption device (along with any other devices inspecting traffic) is passive and not in the flow of traffic, so policies cannot be enforced through it. You also need the private keys of the servers the traffic is intended for, because in this configuration you are not in the middle of the network path, and cannot re-encrypt traffic. This model is only appropriate for inspecting traffic to internal servers, and so not as common.

Inline deployment models effectively position the device as a man-in-the-middle between employees and their website destinations. The decryption device terminates user sessions and establishes new sessions with destination websites. To avoid having the user see a security alert every time they initiate a secure session, users need to trust a certificate issued by the decryption device. That means either loading a signed certificate into every user's browser (typically through a workstation policy) or having decryption device certificates signed by a root which users (browsers) already trust, such as a public Certificate Authority (CA).

Earlier we mentioned the importance of deployment flexibility, which includes being able to assign a certain port as either an active (potentially drop attacks) or passive (only alert on attacks) port sending traffic to an active control. Another port may send traffic passively to a network forensic/security analytics device. And you may want to change those assignments based on dynamic usage models, so the deployment can reflect policies as they evolve.

These devices inspect and influence sensitive traffic so availability is critical, except in passive tap deployments, because taps aren't in the traffic flow. But for an inline configuration you need to decide what happens if the device fails.

Choices include:

1. **Fail to Network:** If the device fails traffic is sent to the outbound network port but not to inspection devices. Sessions are not disrupted, but failure circumvents inspection and monitoring.

2. **Fail to Device:** If the decryption device fails traffic is still sent to inspection devices. Encrypted traffic cannot be decrypted and provides limited means for examination (source, destination, protocol), but unencrypted traffic can still be inspected and monitored/captured.

3. **Fail Closed:** This configuration takes the network offline by not forwarding traffic when the decryption device is down. This ensures you don't miss an attack by taking the network out of commission entirely.

Most organizations choose Fail to Network — if decryption fails they are not willing to stop all traffic. But that is something you will need to figure out with senior management, to ensure they understand the ramifications.

Now that you know how to establish policies to deal with encrypted traffic, where to decrypt it, and the criteria that should guide you in selecting where and how to decrypt, you are ready to deal with encrypted networks.

Summary

Given attacker's increasing sophistication and likelihood they'll gain a foothold on your network, systematically move laterally to their target, package up critical data and exfiltrate it outside your network, what you don't see *can* kill you. So it has become imperative to peek into encrypted data as it comes into and goes out of your network — at least to make sure it's not command and control traffic or leaking critical information.

Yet, decryption can be messy business. First you have to identify the encrypted traffic, then you need to make sure you have appropriate policies to decrypt the sessions at risk, without violating human resources or employee privacy protection mandates. You also need to ensure decryption itself doesn't add undue latency or cause an unacceptable performance impact on your network.

The encrypted network is becoming more of a reality every day, which means the only way to effectively enforce security policies, catch attacks, and monitor networks is to see all traffic.

Organizations have a variety of options to decrypt traffic, including using existing network security devices or deploying a dedicated decryption device. Organizations with modest decryption requirements are generally able to use an existing device, but organizations needing either high performance or deployment flexibility should opt for dedicated devices.

This paper has described how to set policies, determine what type of device will meet your requirements, and select a dedicated decryption device. Since you cannot completely stop attackers, being able to React Faster and Better is the only real option. The encrypted network is now a reality within many organizations, which means the only way to effectively enforce security policies, catch attacks, and monitor networks is to see *all* traffic. The sooner you put an encrypted traffic management strategy and supporting network security architecture in place, the more likely you are to catch your next attacker in the act.

If you have any questions on this topic, or want to discuss your situation specifically, feel free to send us a note at info@securosis.com.

About the Analyst

Mike Rothman, Analyst and President

Mike's bold perspectives and irreverent style are invaluable as companies determine effective strategies to grapple with the dynamic security threatscape. Mike specializes in the sexy aspects of security — such as protecting networks and endpoints, security management, and compliance. Mike is one of the most sought-after speakers and commentators in the security business, and brings a deep background in information security. After 20 years in and around security, he's one of the guys who "knows where the bodies are buried" in the space.

Starting his career as a programmer and networking consultant, Mike joined META Group in 1993 and spearheaded META's initial foray into information security research. Mike left META in 1998 to found SHYM Technology, a pioneer in the PKI software market, and then held executive roles at CipherTrust and TruSecure. After getting fed up with vendor life, Mike started Security Incite in 2006 to provide a voice of reason in an over-hyped yet underwhelming security industry. After taking a short detour as Senior VP, Strategy at eIQnetworks to chase shiny objects in security and compliance management, Mike joined Securosis with a rejuvenated cynicism about the state of security and what it takes to survive as a security professional.

Mike published The Pragmatic CSO <http://www.pragmaticcso.com/> in 2007 to introduce technically oriented security professionals to the nuances of what is required to be a senior security professional. He also possesses a very expensive engineering degree in Operations Research and Industrial Engineering from Cornell University. His folks are overjoyed that he uses literally zero percent of his education on a daily basis. He can be reached at mrothman (at) securosis (dot) com.

About Securosis

Securosis, LLC is an independent research and analysis firm dedicated to thought leadership, objectivity, and transparency. Our analysts have all held executive level positions and are dedicated to providing high-value, pragmatic advisory services. Our services include:

- **Primary research publishing**: We currently release the vast majority of our research for free through our blog, and archive it in our Research Library. Most of these research documents can be sponsored for distribution on an annual basis. All published materials and presentations meet our strict objectivity requirements and conform to our Totally Transparent Research policy.
- **Research products and strategic advisory services for end users**: Securosis will be introducing a line of research products and inquiry-based subscription services designed to assist end user organizations in accelerating project and program success. Additional advisory projects are also available, including product selection assistance, technology and architecture strategy, education, security management evaluations, and risk assessment.
- **Retainer services for vendors**: Although we will accept briefings from anyone, some vendors opt for a tighter, ongoing relationship. We offer a number of flexible retainer packages. Services available as part of a retainer package include market and product analysis and strategy, technology guidance, product evaluation, and merger and acquisition assessment. Even with paid clients, we maintain our strict objectivity and confidentiality requirements. More information on our retainer services (PDF) is available.
- **External speaking and editorial**: Securosis analysts frequently speak at industry events, give online presentations, and write and/or speak for a variety of publications and media.
- **Other expert services**: Securosis analysts are available for other services as well, including Strategic Advisory Days, Strategy Consulting engagements, and Investor Services. These tend to be customized to meet a client's particular requirements.

Our clients range from stealth startups to some of the best known technology vendors and end users. Clients include large financial institutions, institutional investors, mid-sized enterprises, and major security vendors.

Additionally, Securosis partners with security testing labs to provide unique product evaluations that combine in-depth technical analysis with high-level product, architecture, and market analysis. For more information about Securosis, visit our website: <http://securosis.com/>.