PATIENT Side Meeting 2017-11-15

Location: Orchard Room – Rafles City – Singapore
Chair: Brian Witten bwitten@symantec.com, Editor: Arnaud Taddei Arnaud.Taddei.IETF@protonmail.com

Disclaimer and Logistical Issues

- These notes were taken on a best effort basis
- As IETF plenary took an additional hour to finish the chair delayed side meeting start as late as reasonable
- The microphones were actually not working

Presentation

Session PATIENT – Protecting Against Tunnelling in Encrypted Network Traffic – started by the welcome from the Chair, who proposed to run a slide deck as problem statement and defer questions after the presentation.

In summary whilst half of all web connections are encrypted, half of all the web attacks are encrypted too. The presentation then stepped through a refresher on fundamentals and an analysis of the problems, along with some of the alternatives that were considered. The presentation also described protocols that work in parallel and in conjunction with TLS without modifying it. The presentation also reviewed various aspects of this proposal with the chair noting both SDN/NFV on one side as well as trusted execution environments on the other side as potentially helping to facilitate a yet safer solution for everyone. A more detailed description of this presentation is given in the paragraphs below.

As mentioned, with half of web connections now encrypted, half of web attacks are now also encrypted. When Alice talks to a server Bob in the cloud, Alice cannot necessarily trust Bob because Bob may have been compromised and may now be acting maliciously and may try to hack or track Alice – but Alice still might want or need to try to get information from the Bob. Alice may not have enough protection as an endpoint, so she wants to enlist the help of someone in the network to help protect her. The idea is that Alice should be able to determine who in the network she trusts to defend her, and when. To use middleboxes to scan for malicious payloads, the traditional approach was to have a TLS-terminating middlebox, terminating one session with Alice and starting another one on the other side with Bob. Maybe the middlebox uses weaker crypto to talk to Bob than Alice would like, and currently Alice has no way of knowing when that is happening. This limitation of the current approach was described recently in far more detail in "The Security Impact of HTTPS Interception," by Zakir Durumeric, Zane Ma, Drew Springall, Richard Barnes, Nick Sullivan, Elie Bursztein, Michael Bailey, J. Alex Halderman, and Vern Paxson, at Network and Distributed System Security Symposium (NDSS) 2017. In that paper, the risks between different middleboxes varies widely. In this context, users deserve to know more about the risks specific to each middlebox which they are considering to leverage for protection from remote servers. Two alternatives proposed include (a) Alice and Bob communicate directly over TLS but Alice passes keys (symmetric or ephemeral) to the middlebox, or (b) the remote server (as well as upstream middleboxes) report on the connections they have established, although this second option does not eliminate possibility of collusion between upstream middleboxes. mbTLS, mcTLS, TLS-RaR are all various proposals for potential starting points but the chair believed that each such starting point might require refinement before being ready for standardization. Such proposals are not without their problems since, as with current middlebox proxies, they allow violation of end-to-end integrity without accountability of who changed what where when and why. That concern of course can be addressed through adding something akin to Stickler as an additional coordination protocol. However, none of that solves the core concern that the middlebox might be hacked or otherwise accessed by harmful parties. This of course motivates the sensitivities toward (i) carefully choosing when to trust a middlebox in addition to endpoint security instead of depending on just endpoint security, and (ii) when trusting middleboxes, carefully choosing which middlebox to trust.

Values that a middlebox, carefully chosen by a user for their self-protection, and operated by a trustworthy authority trusted by the user, of course include (1) privacy enhancement where a middlebox can hide an end point's information so that they cannot be tracked as easily; (2) preventing an endpoint from being compromised by a remote malicious endpoint, and in special cases (3) potentially supporting stronger crypto, such as perhaps eventually post-quantum crypto, where resource limited endpoints are not able to support such stronger crypto.

Potential solutions include a protocol or combination of protocols that can be used in parallel with TLS, without changes to TLS. Such approaches can work across multiple versions and extensions of TLS. There would be a protocol for Alice to fetch information about the middlebox (what is it, what does it do, how was it made, who's operating it, and other questions pertaining to whether or not it can be trusted) plus protocols like Stickler for better protecting end-to-end integrity, and a protocol for either (a) endpoints sending (symmetric or ephemeral) keys to middleboxes, or (b) endpoints learning the cryptographic strength of upstream sessions & connections. Core principles include, (1) no middlebox would be able to decrypt traffic unless explicitly trusted by an endpoint, and (2) middleboxes would be allowed to delete or change traffic (e.g. to remove malicious ads on a webpage), but there would be a cryptographically strong audit trail of what was changed – a signed manifest of changes. Some versions of such protocols could allow a resource-constrained Alice to let a middlebox choose a stronger cryptographic suite (for example post-quantum) on her behalf. In some cases, this could be done with or without a middlebox terminating Alice's session since conventional crypto could be tunneled inside post-quantum crypto of either an upstream gateway/middlebox or the remote server (Bob) were cooperating in the scheme.

Given the technical limits of the session (microphones not working, meetecho not prepared in time), the chair committed to repeat the contents of the presentation in a WebEx session November 30 at 9am Pacific, details to be announced on the PATIENT mailing list. The chair & proponents are considering requesting a working-group-forming BoF session at IETF 101 in London.

The floor was opened for questions.

Q&A

Question 1 – Would the ephemeral mechanism allow rewrite?
Answer 1 – Yes effective network security requires the ability to rewrite for a number of good reasons, such as removing attacks, and the ephemeral mechanism supports that. See 'Stickler' as a mechanism for integrity protection that would still work for integrity protection or redacted sets of objects. Redaction is needed more granularity than just blocking or dropping a session since sometimes an attack is only a maliciously injected ad or single image, and dropping the whole connection might be an unneeded Denial of Service.

Question 2 – Acronym PATIENT has a T for tunnel? Is it based on client collaboration but need a non malicious client?
Answer 2 – Yes, T is for Tunnel and yes, when we are protecting the client, we assume that the client is not malicious.

Question 3 – Is this about anti-leakage and anti-malware?
Answer 3 – Yes, this is certainly about blocking malware in encrypted tunnels. Of course, inspection can also be applied to leakage detection but our focus is on protecting a cooperating endpoint.

Question 4 – TLS1.3 is making it harder? Is this driven by TLS v1.3?
Answer 4 – No, middleboxes can fully proxy TLS1.3. The problem is that more of the attacks are encrypted, most intrusion prevention systems are going blind to these attacks, forcing more use of proxies, so we want use of proxies to be as safe as possible. It's really being driven by ubiquitous encryption enabled by the likes of Let's encrypt – it's easy for anyone, including phishing websites, and malware hosters to get TLS certificate signed by a CA. It's already possible to proxy TLS v1.3. We would just like to do it in a way that's more controllable by Alice.
Question 4' – Then isn't it that we have a hammer called middleboxes and we want to use it close to the Endpoint. Wouldn't it be better if we can take the middleboxes away?
Answer 4' – It works best when the middlebox is logically close, such as physically close or logically close, such as a VPN, but on why isn't the security on the Endpoint, we certainly want endpoint security to be as strong as possible, and we do tons of endpoint security, but people need layers of security in both the endpoint and in the network, particularly as it makes sense to move away a risk as a grenade from our pocket to a middlebox.
Question 4'' – Isn't it just better to improve the security on the endpoint, rather than rely on network protection?
Answer 4'' – For blocking attacks, a combination of endpoint and network protection is better than endpoint protection alone. For those still disinclined to ever trust middleboxes, preferring to just continue adding endpoint security, a few things are important to note. First, anything that can be done on an endpoint device can be done in a network device, but when things go wrong, it's better for them to go wrong not in the end-user's pocket, but on device or VM somewhere in the cloud, far away from the user, and far away from any credentials or other information they might have stored on their local device. This holds true even for high assurance separation kernels, and formally verified virtualization such as the security enhanced L4 microkernel, and even proprietary implementations of hardware backed separation such as TrustZone and SGX. For nearly every form of virtualization or compartmentalization, escape-ropes either exist, or could be made without the defender's knowledge. When bad things like that happen, it's better for those bad things to happen on a disposable and/or easily reset network appliance and/or container within that appliance, not in the user's

pocket. Second, not every device maker builds in such world class security into the end-devices they make and sell. In fact, the outlook for IOT security is pretty grim, and despite increasing investments and increasing sophistication in the security of mobile operating systems, we've begun to see zero-days in the more secure of the most popular mobile operating systems, and the most popular mobile operating systems not only seem to have a constant strong of vulnerabilities to be patched, but dependencies from the operating system team to the device maker to the carrier cause those patches to reach end-users only very slowly. In contrast, network mitigation is far easier to deploy far faster at far vaster scale across a much greater diversity of devices. We're not saying network in lieu of endpoint. We're saying that serious security requires both better endpoint security, and better network security.

Question 5 – When you move the problem to the cloud, are you not sending sensitive stuff with the grenade?
Answer 5 – You can compartmentalize the risk, sending one moment's stream to one container of one middlebox, and sending the next session to a different container of a different middlebox. That way, the middleboxes don't accrue the valuable jewels over time as does an endpoint. Also, the middlebox isn't physically with me, so that doesn't let users be physically tracked the way an endpoint compromise does. Of course, you still have to trust the middlebox operator, but some people trust their security provider more than the trust the remote server, and sometimes even more than the endpoint device maker.

Question 6 – About the mcTLS and mbTLS which were research projects, aren't we going too fast to move research projects to Standards?
Answer 6 – There are already companies working on those, and implementations of the protocols, with published results, so it is not active research anymore but rather now protocols which could be considered as starting points for standards. Still, the intent is not to immediately ratify one, but rather to develop a consensus on which general approach is preferred, then through refinement and further interop testing via hackathons, direct collaboration, and controlled trials, ensure that the a refined protocol is solid before moving for finalization of an RFC.

Question 7 – About middleboxes and TLS1.3[1]. You are proposing that the client connects to the middlebox sending an ephemeral key but do we need that for a new protocol? Why is it necessary to pass keys to the middlebox, why do we need a new protocol, isn't what is being done today (with proxies) sufficient?
Answer 7 – For the client to get help from the middlebox, it must trust the middlebox, either in allowing it to terminate a TLS session, as it does today, or in sending it either an ephemeral or symmetric secret. I'm happy with either approach as a starting point. The problem today is that the client may not know what the middlebox is doing and it may negotiate a weaker crypt algorithm than the client would allow. In that sense, the client is inheriting more risks than they need to. However, we have multiple solutions available to us. As mentioned earlier, upstream devices could let the endpoint know what downstream devices have done, or we could shift to the model preserving true-remote-server-to-local client sessions, disallowing intermediary negotiation of sessions, with the endpoint controlling distribution of symmetric or ephemeral secrets.

Question 8 – Comment on Endpoints and middleboxes that the premise is right and we need with the right tools

Question 9 – Does the server need to agree to this? Is there any point if the server needs to agree to this? What if the client connects via TLS but the server wants to remove clients with anyone listening and/or protecting them?
Answer 9 – Great scoping question. I believe that either endpoint should be able to get help from the network in protecting against attacks hiding in encrypted tunnels, but that's a great question where we'd want consensus in scoping a working group. You also raise the question of whether the server should be aware of the client trusting a device to shield its privacy. That's of course similarly a great question for scoping or requirements.
Question 9' – Does the server (Bob) participate in this protocol at all?
Answer 9' – As presented today, it's the client seeking protection from the server, and the server only participates in places like Stickler where the client is trying to protect against a potentially malicious middlebox. In those cases, the client has marginal trust for both the middlebox and remote server and uses each to mitigate some of the risks of the other. However, we can imagine other versions of the protocol where the server seeks similar help from network middlboxes. In that sense, it's really a scoping question, and we propose protecting clients first.
Question 9'' - The server might want to hold the client accountable for everything that it does, but if the client gives its keys to a middlebox, then the client might be able to pass blame to the middlebox. Worse, in some US states, both parties have to consent to interception, otherwise it's illegal, but in the proposed solution only one party is consenting. There is no consideration given to the content provider which might have its data intercepted without its consent.

---

[1] Surprise by the room that TLS1.3 is ok and being implemented by middleboxes

Answer 9'' – I'm not a lawyer, but I'm under the impression that if the client empowers a middlebox, of their own, or any other party, then that client is liable for empowering that middlebox, just as the client would be liable for sharing the information with a supplier or other business partner. Still, if we need to separate the "forensic" case, post-facto forensic investigation of advanced threats, and using things like listening & recording of "intercepted" communications to forensically investigate attacks, separating the "forensic" case from the "real-time" case of blocking server to client attacks in real-time, I'm happy to split them. After all, for the forensic case, I believe that the RHRD draft is up for discussion in London. If everyone is satisfied with that for the forensic case, then that leaves us only the real-time case, which more directly protects people. Either way, I'd like to keep the scope focused on middleboxes which an endpoint has chosen to help with protection. That's a strong contrast to hostile monitoring by Law Enforcement and Intelligence Agency interception boxes.

Question 10 – This is not a 2 party but an N party protocol
Answer 10 – Precisely correct.
Question 10' – Given we are talking about endpoints which might not be sophisticated enough, how do you think the endpoint will choose the policy decision? How can uneducated users manage this?
Answer 10' – This can be done several ways, some as simple as with just a VPN and a root of trust, and still protocol refinements could reduce risk. For more complex solutions, users would most likely pick a protection provider to handle the more complicated details for them.

Question 11 – About software updates for IoT?
Answer 11 – Of course, now IoT is complicated because of the supply chain. Lots of smartphones and cars don't get updates fast enough. Often it's easier to keep a few gateways up to date for protecting many such devices.

Question 12 – Go back: the Endpoint can be on a managed network or an unmanaged network since there is a need for a policy is there some kind of discovery?
Answer 12 – There are static and dynamic ways to look at the discovery, but I'd be happy to keep the scope limited to endpoints trusting only a pre-configured set of gateways if that's the consensus of the group on scoping the first version of such a protocol.

Question 13 – What the solution look like for the user, what is the advantage for the browser? What is the user consent? It comes down to control on who? It seems that this is a power play shifting the emphasis of control to someone else? How to maintain the social contract?
Answer 13 – Users have a right to protect themselves, including choosing to have their communications mediated by a security provider but the point is the client should be able to control and choose whether or not such communications are mediated.

Question 14 – Just a VPN & locally installed root?
Answer 14 – Today this is all or nothing. We'd like to standardize something that gives endpoints much more options and much more control in how, when, and where they get such security help from the network. We'd also like to standardize a way for the endpoints to verify that the middlebox is doing its job properly, just as the middlebox helps verify that the server is safe. In both cases, the model is "trust but verify."

Question 15 – As a simple TLS caveman or TLS expert. I see no cases on the browser for such a need.
Answer 15 – Today the market is like this, but browsers don't adequately block phishing sites, malware downloads, and other privacy risks to end users including server side profiling and tracking of clients. Also, I'm thrilled to see so much progress in browser security, but no software, including browsers are ever perfect. Each eventually have vulnerabilities discovered, sometimes privately before publicly. These are all among the reasons enterprise customers already use network security to protect their endpoints. The same need is coming for most consumers, it's just not mass market yet.

Question 16 – Comment about the fact that there is a fraction that are choosing their own security so there is a case for that today

Question 17 – Discussion about the Red Screen (invalid cert), some people want to have install their CA because of Parental control, etc. or others have VPN solutions which fail a lot because VPNs are blocked
Answer 17 – We try to give users a better solution here. Of course, better solutions are available, and yet better solutions could exist with the right standards giving users more insight and more control.

Question 18 – Not all TLS traffic comes from Browsers, there are many situations where you benefit from network security.
Answer 18 – Thanks, agree completely. Other important examples include IOT devices like connected cars, and mobile apps.

Question 19 – People connect to their VPN because of email, on custom roots they bought the Marketing spin.
Answer 19 – These boxes block countless attacks daily. As a company, we block millions of attacks daily, roughly half of them detected with network facing engines, and as the network goes dark from pervasive encryption, that dramatically increases the users risks. Middleboxes help mitigate that risk, but if as a community we're going to put so much onto the middleboxes, we want to do all possible to make them as safe as possible.
Question 19' - What about the UX? Isn't it going to be confusing?
Answer 19' – Part of the value is that middleboxes give users a choice on whether or not to trust the remote server almost completely and almost completely blindly, or to get some serious risk mitigation in watching what the server is trying to do the client. Obviously for a solution to sell well and be effective at scale, it must give the right choices to the user without confusing them. Today it is not clear to the user when they are or are not being protected and/or monitored. We should change that so that users can have more protection against malicious servers and more protection against middleboxes that are incompetent or potentially malicious. Of course, if the UX is terrible, then consumer middlebox services won't sell. In the Enterprise context, we could still make it far easier for the user to know if, why, when, where, and how their communications are being monitored. That's particularly important as most employers have already opted out of monitoring employee Personal Financial & Healthcare communications. That's also increasingly important as the number of mixed use "work and play" devices continues growing through trends like BYOD. We agree that getting the user experience right is crucial. However that is mostly on the security service (middlebox) provider since their boxes & services won't sell as well if the UX is terrible.

Question 20– Thank you for keeping this conversation and indeed balancing act between the endpoint and the network
Answer 20 – Thank you again!
Answer 20 (Editor) – Sharing on the long debate with Digital Service Providers on where to put security: endpoint or network? Long and big disagreement for years until proposal to engage peace talks and definition of an Hybrid Security Architecture between Endpoint and Middleboxes. but still obviously it is a Hot debate.

Question 21 – We heard about the control, browsers, IoT, corporate machines, etc. Is it the focus on Enterprises or on Users?
Answer 21 – We want to focus on the user, the user control the U/X and we need to make it simple but this is about the user to start.

Question 22 – We hear about trust and control but when it comes to trust we have already many parties we need to trust: the browser, the OS, the App, the opensource. It is not a user here this is not a personal relationship. There are many layers of trust already. Sometimes people trust the network or their security provider more than some of these other parties.
Answer 22: Great point thank you, I agree completely.

Question 23 (?) – There is TLS1.3, and if middleboxes can proxy that, why are we here?
Answer 23 – Attacks are tunneled through TLS 1.2 as well as 1.3, and people increasingly need middleboxes as a way to protect themselves against these attacks, particularly for mobile & IoT devices, as well as classic devices. Why IETF is because getting endpoints and middleboxes to work together better and more safely will require new internet standards in parallel to TLS.

Question 24 – Commented that the Hybrid Security Architecture is about sharing profit with the Operators, it's about profit.
Answer 24 (Editor) – Push back on this is an unfair interpretation.
Answer 24 – This is about preventing attacks like phishing attacks and protecting users.
Side-comment: allegations of profit focus are not professional. Nobody is alleging that advertisers and content providers are against middleboxes as middleboxes could strip out ads hurting revenues. We are all in this to best protect people. Some people believe that people are best protected with end-to-end crypto without any inspection in the network, other people believe that potentially dangerous traffic should be inspected before it hits a potentially vulnerable endpoint, even if that traffic is encrypted. Both sides bring decades of individual experience and millenia of collective experience to this discussion which might very well be the single most important security architecture question of the next twenty years, made urgent now as LetsEncrypt finally drives ubiquitous encryption. Now that we're getting ubiquitous encryption, we need to get it right. 1.3 is a huge part of that. Blocking the attacks in the tunnels is the next, and perhaps in many ways, even more crucial part of that.

Question 25 – I'd like to see the data, verifiable data, and the evidence assembled as an Internet Draft.
Answer 25 – Great request. Happy to do that. Will aim to do so before London.

Question 26 – What does success look like from a browser vendor perspective? Look for a more concrete proposal think about children, think about business model, why is it uniquely positioned and how do I talk about it to my users? Opened to ideas but fundamental problems
Answer 26 – More Server to client attacks blocked before they reach the browser, and the browser better able to diagnose who has tampered with content coming from the server, particularly as middleboxes already re-write content today for several reasons, even in encrypted tunnels.

Question 27 – I don't see any value on this, I never installed any anti-malware on my device and I do go to all the bad sites.
Answer 27 – Unfortunately, the thousands of security experts ready for such a discussion represent less than 00.02% of the people on the planet, and I've tried getting doctors, teachers, nurses, and caregivers to practice great op-sec, to use locked down versions of Kali linux, and I didn't get very far with that. This seems to scale way better.

Question 28 – How do I recognize a MITM vs a Middlebox?
Answer 28 – Need to make a conscience decision early to trust one middlebox operator and not randomly start trusting others. Of course, this raises scoping questions like "do we need discovery? Should we avoid discovery?" As mentioned above, I'm happy to avoid discovery for early versions of this protocol. A consumer friendly example of suitable simplicity would be current approaches to Wifi Privacy. Installing such an app is a very conscious decision, and could easily additionally include a root of trust for blocking server to client attacks.

Question 29– Glad you mentioned the Middle East attack on iOS and the zero days but I came to the conclusion that the middleboxes are a problem. This is personal for many of us as one of my friend disappeared! If you trust the wrong middlebox, they get you.
Answer 29 – That's exactly why we're here. Middleboxes can protect against the Zero Days, but we need to make it easier to be sure you can trust the middleboxes, and to know which middlebox operator, and to know the regime in which it's operating, subpoena risks and all of those crucial details. Not only do we want to make the standards better for this, but should this approach work for the community, when it will come to productization, as a company, we're even considering giving instances of our software to a non profit organization for them to run it themselves for such situations where people can only trust a non-profit to protect them.

Question 30 – Discussion on email is it the primary vector of attack?
Answer 30 – As mentioned, we'll pull together the data for this into an Internet Draft (Question 25)

Question 31 – What stops you to monetize more? I want to minimize the number of places to trust
Answer 31 – I see, you are concerned that the security provider might start selling customer profiles instead of just preventing the advertisers from profiling their customers. I believe that the success of anyone selling privacy protection services will depend on how they protect their brand and reputation for protecting privacy.

Some take-aways:

- Focus on consumer endpoints should not exclude enterprise endpoints.
- Enterprise endpoints should stay explicitly in scope.
- Real-time attack blocking case is certainly in scope.
- Forensic attack analysis would be in scope if the RHRD solution is not accepted in London.
- Data Loss/Leakage Prevention is not explicitly in scope. The user is trusted. The end-user's endpoint is trusted.
- Risks to be mitigated include BOTH potentially malicious servers & potentially malicious middleboxes.
- It could be in scope for the most reputable servers to know whether or not a middlebox is protecting an endpoint. It would not be appropriate for potentially malicious servers to be able to insist on that information.

After The Close

The chair closed the mic and asked for more questions especially about a BOF in London. He validated with the team feedback that people attended this Bar BOF because they do care.

Someone for fairness reformulated the hum which was turned into hands raised as:

Do we think it is an area where standardization is crucial?

The result was a 50% 50% that IETF should or should not engage here

It was recognized that we are in fact talking about a Network Security Function (NSF as in i2nsf) and how the user can declare their security. It was mentioned too that in some states (and in some US states laws) one party agreement is not sufficient.

Several debates and discussions started offline on the rationale for a more specific definition of endpoint and a signalable & mutually agreed model.

Meeting was closed with a round of applause that we could discuss together even with very different opinions

Statistics

- The room was full of around 60 people
- With several in and out it was evaluated around 90 people attended.

References

[1] Stickler: Defending Against Malicious CDNs in an Unmodified Browser
www.amitlevy.com/papers/stickler-w2sp15.pdf
[2] Housley, R and Droms, R. TLS 1.3 Option for Negotiation of Visibility in the Datacenter draft-rhrd-tls-tls13-visibility-00. September 29th of 2017. In https://datatracker.ietf.org/doc/draft-rhrd-tls-tls13-visibility/