Network Working Group
Internet-Draft
Intended status: Informational
Expires: 13 January 2022

J. Arkko
Ericsson
S. Farrell
Trinity College Dublin

July 2021

Internet Threat Model Guidance

draft-arkko-arch-internet-threat-model-guidance-00.txt

Abstract

Communications security has been at the center of many security improvements in the Internet. The goal has been to ensure that communications are protected against outside observers and attackers.

This memo suggests that the existing RFC 3552 threat model, while important and still valid, is no longer alone sufficient to cater for the pressing security and privacy issues seen on the Internet today. For instance, it is often also necessary to protect against endpoints that are compromised, malicious, or whose interests simply do not align with the interests of users. While such protection is difficult, there are some measures that can be taken and we argue that investigation of these issues is warranted.

It is particularly important to ensure that as we continue to develop Internet technology, non-communications security related threats, and privacy issues, are properly understood.

Copyright Notice

Table of Contents

1. Introduction

Communications security has been at the center of many security improvements in the Internet. The goal has been to ensure that communications are protected against outside observers and attackers. At the IETF, this approach has been formalized in BCP 72 [RFC3552], which defined the Internet threat model in 2003.

The purpose of a threat model is to outline what threats exist in order to assist the protocol designer. But RFC 3552 also ruled some

threats to be in scope and of primary interest, and some threats out of scope [RFC3552]:

> The Internet environment has a fairly well understood threat model. In general, we assume that the end-systems engaging in a protocol exchange have not themselves been compromised. Protecting against an attack when one of the end-systems has been compromised is extraordinarily difficult. It is, however, possible to design protocols which minimize the extent of the damage done under these circumstances.
>
> By contrast, we assume that the attacker has nearly complete control of the communications channel over which the end-systems communicate. This means that the attacker can read any PDU (Protocol Data Unit) on the network and undetectably remove, change, or inject forged packets onto the wire.

However, the communications-security -only threat model is becoming outdated. Some of the causes for this are:

* Success! Advances in protecting most of our communications with strong cryptographic means. This has resulted in much improved communications security, but also highlights the need for addressing other, remaining issues. This is not to say that communications security is not important, it still is. Fortunately, there are ongoing projects working on improvements.

* Adversaries have increased their pressure against other avenues of attack, from supply-channel attacks, to compromising devices to legal coercion of centralized endpoints in conversations.

* New adversaries and risks have arisen, e.g., due to creation of large centralized information sources.

* While communications-security does seem to be required to protect privacy, more is needed, especially if endpoints choose to act against the interests of their peers or users.

In short, attacks are migrating towards the currently easier targets, which no longer necessarily include direct attacks on traffic flows.

It is important that when it comes to basic Internet infrastructure, our chosen technologies lead to minimal exposure with respect to the non-communications threats. It is particularly important to ensure that non-communications security related threats are properly understood for any new Internet technology. The sole consideration of communications security aspects in designing Internet protocols may lead to accidental or increased impact of security issues

elsewhere. For instance, allowing a participant to unnecessarily collect or receive information may lead to a similar effect as described in [RFC8546] for protocols: over time, unnecessary information will get used with all the associated downsides, regardless of what deployment expectations there were during protocol design.

The rest of this memo is organized as follows. Section 2 makes some observations about the threa situation. Section 3 discusses some high-level principles that could be used to address some of the emerging issues.

2. Attack Landscape

This section discusses the evolving landscape of security vulnerabilities, threats, and attacks.

2.1. Communications Security Improvements

Being able to ask about threat model improvements is due to progress already made: The fraction of Internet traffic that is cryptographically protected has grown tremendously in the last few years. Several factors have contributed to this change, from Snowden revelations to business reasons and to better available technology such as HTTP/2 [RFC7540], TLS 1.3 [RFC8446], QUIC [I-D.ietf-quic-transport].

In many networks, the majority of traffic has flipped from being cleartext to being encrypted. Reaching the level of (almost) all traffic being encrypted is no longer something unthinkable but rather a likely outcome in a few years.

At the same time, technology developments and policy choices have driven the scope of cryptographic protection from protecting only the pure payload to protecting much of the rest as well, including far more header and meta-data information than was protected before. For instance, efforts are ongoing in the IETF to assist encrypting transport headers [I-D.ietf-quic-transport], server domain name information in TLS [I-D.ietf-tls-esni], and domain name queries [RFC8484].

There have also been improvements to ensure that the security protocols that are in use actually have suitable credentials and that those credentials have not been compromised, see, for instance, Let's Encrypt [RFC8555], HSTS [RFC6797], HPKP [RFC7469], and Expect-CT [I-D.ietf-httpbis-expect-ct].

This is not to say that all problems in communications security have been resolved -- far from it. But the situation is definitely different from what it was a few years ago. Remaining issues will be and are worked on; the fight between defense and attack will also continue. Communications security will stay at the top of the agenda in any Internet technology development.

2.2. Beyond Communications Security

There are issues beyond communications security in the Internet. It is not necessarily clear that one can trust all the endpoints in any protocol interaction, including the user's own devices. Managed or closed ecosystems with multiple layers of hardware and software have made it harder to understand or influence what your devices do.

The situation is different, but not necessarily better on the side of servers. Even for applications that are for user-to-user communication, a typical pattern of communications is almost always via an intermediary that has at least as much information as the other parties have. For instance, these intermediaries are typically endpoints for any transport layer security connections, and able to see much communications or other messaging in cleartext. There are some exceptions, of course, e.g., messaging applications with end-to-end confidentiality protection.

For instance, while e-mail transport security [RFC7817] has become much more widely deployed in recent years, progress in securing e-mail messages between users has been much slower. This has lead to a situation where e-mail content is considered a critical resource by some mail service providers who use the content for machine learning, advertisement targeting, and other purposes unrelated to message delivery. Equally however, it is unclear how some useful anti-spam techniques could be deployed in an end-to-end encrypted mail universe (with today's end-to-end mail security protocols) and there are many significant challenges should one desire to deploy end-to-end email security at scale.

Even services that are not about user-to-user to communication often collect information about the user.

Services that are part of the infrastructure may have security issues. For instance, despite progress in protecting DNS query protocols with encryption (see, e.g., [RFC7858] and [RFC8484]), DNS resolver services themselves may be targets for attack or sources for leaks. For instance, the services may collect information or be vulnerable targets of denial-of-service attacks, attacks to steal user browsing history information, or be the target of surveillance activities and government information requests. Infrastructure

services with information about a large number of users is collected in small number of services are particularly attractive targets for these attacks.

In general, many recent attacks relate more to information than communications. For instance, personal information leaks typically happen via information stored on a compromised server rather than capturing communications. There is little hope that such attacks can be prevented entirely. Again, the best course of action seems to be avoid the disclosure of information in the first place, or at least to not perform that in a manner that makes it possible that others can readily use the information.

2.3. Accidental Vulnerabilities

Some vulnerabilities came to being through various levels of carelessness and/or due to erroneous assumptions about the environments in which those applications currently run at. A vulnerability can be exploited to misuse the data for someone's own purposes.

Some attacks in this category include hardware-related issues, for example, Meltdown and Spectre [MeltdownAndSpectre], compromised or badly-maintained web sites or services, e.g., [Passwords], supply-chain attacks, for example, the [TargetAttack], and breaches of major service providers, that many of us might have assumed would be sufficiently capable to be the best large-scale "Identity providers", for example, Yahoo (https://www.wired.com/story/yahoo-breach-three-billion-accounts/), Facebook (https://www.pcmag.com/news/367319/facebook-stored-up-to-600m-user-passwords-in-plain-text and many others.

2.4. Misbehaving Applications

There are many examples of application developers doing their best to protect the security and privacy of their users or customers. But there are also some that do not act int he best interests of their users. As a result, it becomes necessary to consider applications as potentially untrusted, much in the same way that we consider in-network actors as potential adversaries despite the many examples of network operators who both act in the best interests of their users and succeed in defending against attacks from others.

This can also happen indirectly. Despite the best efforts of curators, so-called App-Stores frequently distribute malware of many kinds.

Applications may also mislead users. Many web sites today provide some form of privacy policy and terms of service, that are known to be mostly unread [Unread]. This implies that, legal fiction aside, users of those sites have not in reality agreed to the specific terms published and so users are therefore highly exposed to being exploited by web sites, for example [Cambridge] is a recent well-publicised case where a service provider abused the data of 87 million users via a partnership.

2.5. Untrustworthy Devices

Traditionally, there's been an implied trust in various parts of the system -- such as the user's own device, nodes inside a particular network perimeter, or nodes under a single administrative control.

Client endpoint implementations were never fully trusted, but the environments in which those endpoints exist are changing. Users may not have as much control over their own devices as they used to, due to manufacturer-controlled operating system installations and locked device ecosystems. And within those ecosystems, even the applications that are available tend to have privileges that users by themselves might not desire those applications be granted, such as excessive rights to media, location, and peripherals. There are also designated efforts by various authorities to hack end-user devices as a means of intercepting data about the user.

Examples of these issues are too many to list, for instance, so-called "smart" televisions spying on their owners and one survey of user attitudes [SmartTV]. Untrustworthy devices can also cause damage to other parties, such as badly constructed IoT devices [DynDDoS] attacking large Internet services.

2.6. Untrustworthy "Closed" Networks

Even in a closed network with carefully managed components there may be compromised components, as evidenced in the most extreme way by the Stuxnet worm that operated in an airgapped network. Every system runs large amount of software, and it is often not practical or even possible to prevent compromised code even in a high-security setting, let alone in commercial or private networks. Installation media, physical ports, both open source and proprietary programs, firmware, or even innocent-looking components on a circuit board can be suspect [TinyChip]. In addition, complex underlying computing platforms, such as modern CPUs with underlying security and management tools are prone to problems.

2.7. Tracking

One of the biggest threats to user privacy on the Web is ubiquitous tracking. This is often done to support advertising based business models, or more specifically advertising based business models that attempt to find out information about the user.

While some people may be sanguine about this kind of tracking, others consider this behaviour unwelcome, when or if they are informed that it happens. Historically, browsers have not made this kind of tracking visible and have enabled it by default, though some recent browser versions are starting to enable visibility and blocking of some kinds of tracking.

One form of tracking is by third parties. HTTP header fields (such as cookies, [RFC6265]) are commonly used for such tracking, as are structures within the content of HTTP responses such as links to 1x1 pixel images and (ab)use of Javascript APIs offered by browsers [Tracking]. Whenever a resource is loaded from a server, that server can include a cookie which will be sent back to the server on future loads. The combination of these features makes it possible to track a user across the Web.

This capability itself constitutes a major threat to user privacy. Additional techniques such as cookie syncing, identifier correlation, and fingerprinting make the problem even worse [I-D.wood-pearg-website-fingerprinting]. For instance, features such as User-Agent string, plugin and font support, screen resolution, and timezone can yield a fingerprint that is sometimes unique to a single user [AmIUnique] and which persists beyond cookie scope and lifetime.

Third party web tracking is not the only concern. An obvious tracking danger exists also in popular ecosystems -- such as social media networks -- that house a large part of many users' online existence. There is no need for a third party to track the user's browsing as all actions are performed within a single site, where most messaging, viewing, and sharing activities happen.

Browsers themselves or services used by the browser can also become a potential source of tracking users. For instance, the URL/search bar service may leak information about the user's actions to a search provider via an "autocomplete" feature. [Leith2020]

Tracking through users' IP addresses or DNS queries is also a danger. This may happen by directly observing the cleartext IP or DNS traffic, though DNS tracking may be preventable via DNS protocols that are secured end-to-end. But the DNS queries are also (by

definition) seen by the used DNS recursive resolver service, which may accidentally or otherwise track the users' activities.

Tracking happens through other systems besides the web, of course. For instance, some mail user agents (MUAs) render HTML content by default (with a subset not allowing that to be turned off, perhaps particularly on mobile devices) and thus enable the same kind of adversarial tracking seen on the web.

One of the concerns with universal user tracking is that it provides yet another avenue for pervasive surveillance [RFC7258], e.g., intelligence agencies can tap into the databases constructed by user tracking.

3. Principles

Based on the above issues, it is necessary to pay attention to the following aspects:

* Security of devices, including the user's own devices.

* Security of data at rest or in use, in various parts of the system.

* Tracking and identification of users and their devices.

* Role of servers, and in particular information passing through them.

These topics are discussed below. There are obviously many detailed technical questions and approaches to tackling them. However, in this memo we wish to focus on higher level architectural principles that might guide us in thinking about about the topics.

3.1. Trusting Devices

In general, this means that one cannot entirely trust even a closed system where you picked all the components yourself, let alone typical commercial, networked and Internet-connected systems.

PRINCIPLE: Consider all system components as potentially untrustworthy, and consider the implications of their compromise.

There may also be ways to mitigate damages, should a compromise occur.

3.2. Protecting Information

Data leaks have become the primary concern. Even trusted, well-managed parties can be problematic, such as when large data stores attract attempts to use that data in a manner that is not consistent with the users' interests.

Mere encryption of communications is not sufficient to protect information.

PRINCIPLE: Consider information passed to another party as a publication. Avoid passing information that should not be published.

This principle applies even if the communications that carry that information are encrypted.

PRINCIPLE: Build defences to protect information, even when some component in a system is compromised.

For instance, encryption of data at rest or in use may assist in protecting information when an attacker gainst access to a server system.

PRINCIPLE: Trust that information is handled appropriately, but verify that this is actually the case.

It is not wise to merely trust that someone acts correctly, without mistakes, and does not misuse your information. When we send packets over the Internet, we encrypt them and know that they can only received by a specific party. Similarly, if we send information to a server, we can, for instance:

* encrypt a message only to the actual final recipient, even if the server holds our message before it is delivered

* verify (e.g., through confidential computing attestations) that the server runs a software that we know does not leak our information

3.3. Tracking

Information leakage is particularly harmful in situations where the information can be traced to an individual, such as is the case with any information that users would consider private, be it about messages to another users, browsing history, or even user's medical information.

PRINCIPLE: Assume that every interaction with another party can result in fingerprinting or identification of the user in question.

In many cases there are readily available user identifiers in data that is leaked. But even when such identifiers are not present, there is often an opportunity to narrow down which entity is connecting, through, for instance, geolocation or fingerprinting.

3.4. Role of End-to-End

[RFC1958] noted that "end-to-end functions can best be realised by end-to-end protocols". This functional argument aligns with other, practical arguments about the evolution of the Internet under the end-to-end model. The endpoints evolve quickly, often with simply having one party change the necessary software on both ends. The end-to-end model supports permissionless innovation where new innovation can flourish in the Internet without excessive wait for other parties to act.

However, there is a significant difference between actual endpoints from a user's interaction perspective (e.g., another user) and from a system perspective (e.g., a party relaying a message). In general, there needs to be distinction between the intended interaction participants and the services used to carry this interaction out. These services are typically implemented as servers that provide, e.g., the messaging relay function.

Thomson [I-D.thomson-tmi] discusses the role of intermediaries. We prefer to use the term services to underline how all types of services can have issues -- including the simple case of an end-user contacting a server for some information.

In any case, as Thomson points out, intermediaries (or services) can provide a useful function. Networks themselves would not exist without intermediaries that can forward communications to others. Similarly, networks would not exist without services responding to communications sent by end users.

PRINCIPLE: Set clear limits what all services can do, and minimise the use of those services to cases where they are necessary.

This is a general rule, but perhaps a few examples can illustrate it:

* A router's role is to efficiently forward packets to their destination, not to differentiate the treatment based on what content is being carried.

* The role of an information service web server is to provide that information, not to gather the identity or personal information about the user accessing information.

* The role of a messaging service is to deliver messages to other users, not to process the contents of the messages.

Note that this principle applies at multiple layers in the stack. It is not just about intermediaries in the network and transport layers, but also intermediaries and services on the application layer.

PRINCIPLE: Pass information only between the "real ends" of a conversation, unless the information is necessary for a useful function in a service.

For instance, a transport connection between two components of a system is not an end-to-end connection even if it encompasses all the protocol layers up to the application layer. It is not end-to-end, if the information or control function it carries actually extends beyond those components. For instance, just because an e-mail server can read the contents of an e-mail message does not make it a legitimate recipient of the e-mail.

PRINCIPLE: Information should not be disclosed, stored, or routed in cleartext through services that do not absolutely need to have that information for the function they perform.

This the "need to know" principle. It also relates to the discussion in [I-D.thomson-tmi], in that the valuable functions provided by intermediaries need to be balanced against the information that they need to perform their function.

4. Security Considerations

The entire memo covers the security considerations.

5. IANA Considerations

There are no IANA considerations in this work.

6. Informative References

[AmIUnique]
INRIA, ., "Am I Unique?", https://amiunique.org , 2020.

[Cambridge]
Isaak, J. and M. Hanna, "User Data Privacy: Facebook, Cambridge Analytica, and Privacy Protection", Computer

51.8 (2018): 56-59, https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8436400 , 2018.

[DynDDoS] York, K., "Dyn's Statement on the 10/21/2016 DNS DDoS Attack", Company statement: https://dyn.com/blog/dyn-statement-on-10212016-ddos-attack/ , 2016.

[I-D.arkko-arch-dedr-report]
Arkko, J. and T. Hardie, "Report from the IAB workshop on Design Expectations vs. Deployment Reality in Protocol Development", Work in Progress, Internet-Draft, draft-arkko-arch-dedr-report-00, 4 November 2019, <https://www.ietf.org/archive/id/draft-arkko-arch-dedr-report-00.txt>.

[I-D.arkko-arch-internet-threat-model]
Arkko, J., "Changes in the Internet Threat Model", Work in Progress, Internet-Draft, draft-arkko-arch-internet-threat-model-01, 8 July 2019, <https://www.ietf.org/archive/id/draft-arkko-arch-internet-threat-model-01.txt>.

[I-D.arkko-farrell-arch-model-t]
Arkko, J. and S. Farrell, "Challenges and Changes in the Internet Threat Model", Work in Progress, Internet-Draft, draft-arkko-farrell-arch-model-t-04, 13 July 2020, <https://www.ietf.org/archive/id/draft-arkko-farrell-arch-model-t-04.txt>.

[I-D.arkko-farrell-arch-model-t-redux]
Arkko, J. and S. Farrell, "Internet Threat Model Evolution: Background and Principles", Work in Progress, Internet-Draft, draft-arkko-farrell-arch-model-t-redux-01, 22 February 2021, <https://www.ietf.org/archive/id/draft-arkko-farrell-arch-model-t-redux-01.txt>.

[I-D.farrell-etm]
Farrell, S., "We're gonna need a bigger threat model", Work in Progress, Internet-Draft, draft-farrell-etm-03, 6 July 2019, <https://www.ietf.org/archive/id/draft-farrell-etm-03.txt>.

[I-D.ietf-httpbis-expect-ct]
Stark, E., "Expect-CT Extension for HTTP", Work in Progress, Internet-Draft, draft-ietf-httpbis-expect-ct-08, 9 December 2018, <https://www.ietf.org/archive/id/draft-ietf-httpbis-expect-ct-08.txt>.

[I-D.ietf-quic-transport]
Iyengar, J. and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport", Work in Progress, Internet-Draft, draft-ietf-quic-transport-34, 14 January 2021, <https://www.ietf.org/archive/id/draft-ietf-quic-transport-34.txt>.

[I-D.ietf-tls-esni]
Rescorla, E., Oku, K., Sullivan, N., and C. A. Wood, "TLS Encrypted Client Hello", Work in Progress, Internet-Draft, draft-ietf-tls-esni-12, 7 July 2021, <https://www.ietf.org/archive/id/draft-ietf-tls-esni-12.txt>.

[I-D.lazanski-smart-users-internet]
Lazanski, D., "An Internet for Users Again", Work in Progress, Internet-Draft, draft-lazanski-smart-users-internet-00, 8 July 2019, <https://www.ietf.org/archive/id/draft-lazanski-smart-users-internet-00.txt>.

[I-D.thomson-tmi]
Thomson, M., "Principles for the Involvement of Intermediaries in Internet Protocols", Work in Progress, Internet-Draft, draft-thomson-tmi-02, 6 July 2021, <https://www.ietf.org/archive/id/draft-thomson-tmi-02.txt>.

[I-D.wood-pearg-website-fingerprinting]
Goldberg, I., Wang, T., and C. A. Wood, "Network-Based Website Fingerprinting", Work in Progress, Internet-Draft, draft-wood-pearg-website-fingerprinting-00, 4 November 2019, <https://www.ietf.org/archive/id/draft-wood-pearg-website-fingerprinting-00.txt>.

[Leith2020]
Leith, D., "Web Browser Privacy: What Do Browsers Say When They Phone Home?", In submission, https://www.scss.tcd.ie/Doug.Leith/pubs/browser_privacy.pdf , March 2020.

[MeltdownAndSpectre]
CISA, ., "Meltdown and Spectre Side-Channel Vulnerability Guidance", Alert (TA18-004A), https://www.us-cert.gov/ncas/alerts/TA18-004A , 2018.

[Passwords]

com, haveibeenpwned., "Pwned Passwords", Website https://haveibeenpwned.com/Passwords , 2019.

[RFC1958] Carpenter, B., Ed., "Architectural Principles of the Internet", RFC 1958, DOI 10.17487/RFC1958, June 1996, <https://www.rfc-editor.org/info/rfc1958>.

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003, <https://www.rfc-editor.org/info/rfc3552>.

[RFC6265] Barth, A., "HTTP State Management Mechanism", RFC 6265, DOI 10.17487/RFC6265, April 2011, <https://www.rfc-editor.org/info/rfc6265>.

[RFC6797] Hodges, J., Jackson, C., and A. Barth, "HTTP Strict Transport Security (HSTS)", RFC 6797, DOI 10.17487/RFC6797, November 2012, <https://www.rfc-editor.org/info/rfc6797>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <https://www.rfc-editor.org/info/rfc7258>.

[RFC7469] Evans, C., Palmer, C., and R. Sleevi, "Public Key Pinning Extension for HTTP", RFC 7469, DOI 10.17487/RFC7469, April 2015, <https://www.rfc-editor.org/info/rfc7469>.

[RFC7540] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext Transfer Protocol Version 2 (HTTP/2)", RFC 7540, DOI 10.17487/RFC7540, May 2015, <https://www.rfc-editor.org/info/rfc7540>.

[RFC7817] Melnikov, A., "Updated Transport Layer Security (TLS) Server Identity Check Procedure for Email-Related Protocols", RFC 7817, DOI 10.17487/RFC7817, March 2016, <https://www.rfc-editor.org/info/rfc7817>.

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/info/rfc7858>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <https://www.rfc-editor.org/info/rfc8484>.

[RFC8546] Trammell, B. and M. Kuehlewind, "The Wire Image of a Network Protocol", RFC 8546, DOI 10.17487/RFC8546, April 2019, <https://www.rfc-editor.org/info/rfc8546>.

[RFC8555] Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", RFC 8555, DOI 10.17487/RFC8555, March 2019, <https://www.rfc-editor.org/info/rfc8555>.

[SmartTV] Malkin, N., Bernd, J., Johnson, M., and S. Egelman, "What Can't Data Be Used For? Privacy Expectations about Smart TVs in the U.S.", European Workshop on Usable Security (Euro USEC), https://www.ndss-symposium.org/wp-content/uploads/2018/06/eurousec2018_16_Malkin_paper.pdf" , 2018.

[TargetAttack] Osborne, C., "How hackers stole millions of credit card records from Target", ZDNET, https://www.zdnet.com/article/how-hackers-stole-millions-of-credit-card-records-from-target/ , 2014.

[TinyChip] Robertson, J. and M. Riley, "The Big Hack: How China Used a Tiny Chip to Infiltrate U.S. Companies", https://www.bloomberg.com/news/features/2018-10-04/the-big-hack-how-china-used-a-tiny-chip-to-infiltrate-america-s-top-companies , October 2018.

[Tracking] Ermakova, T., Fabian, B., Bender, B., and K. Klimek, "Web Tracking-A Literature Review on the State of Research", Proceedings of the 51st Hawaii International Conference on System Sciences, https://scholarspace.manoa.hawaii.edu/bitstream/10125/50485/paper0598.pdf , 2018.

[Unread] Obar, J. and A. Oeldorf, "The biggest lie on the internet{:} Ignoring the privacy policies and terms of service policies of social networking services", Information, Communication and Society (2018): 1-20 , 2018.

Appendix A. Contributors

Eric Rescorla and Chris Wood provided much of the text in Section 2.7. Martin Thomson's excellent document [I-D.thomson-tmi] also inspired some of the work in Section 3.

Earlier variations of this draft were produced in [I-D.farrell-etm] [I-D.arkko-arch-internet-threat-model] [I-D.arkko-farrell-arch-model-t] [I-D.arkko-farrell-arch-model-t-redux].

There are also other documents discussing this overall space, e.g. [I-D.lazanski-smart-users-internet] [I-D.arkko-arch-dedr-report].

Appendix B. Acknowledgements

The authors would like to thank the members of the IAB, the participants of the IETF SAAG meeting where this topic was discussed, the participants of the IAB 2019 DEDR workshop, and the participants of the Model-T meetings at the IETFs.

Authors' Addresses

Jari Arkko
Ericsson
Valitie 1B
FI- Kauniainen
Finland

Email: jari.arkko@piuha.net

Stephen Farrell
Trinity College Dublin
College Green
Dublin
Ireland

Email: stephen.farrell@cs.tcd.ie