Network Working Group
Internet-Draft
Intended status: Informational
Expires: October 30, 2014

S. Farrell
Trinity College, Dublin
R. Wenning
B. Bos
W3C
M. Blanchet
Viagenie
H. Tschofenig
ARM Ltd.
April 28, 2014

STRINT workshop report
draft-iab-strint-report-00

Abstract

The STRINT workshop assembled one hundred participants in London for two days in early 2014 to discuss how the technical community, and in particular the IETF and the W3C, should react to Pervasive Monitoring and more generally how to strengthen the Internet in the face of such attacks. The discussions covered issues of terminology, the role of user interfaces, classes of mitigation, some specific use cases, transition strategies (including opportunistic encryption), and more. The workshop ended with a few high-level recommendations, which it is believed could be implemented and which could help strengthen the Internet. This is the report of that workshop.

Table of Contents

1. Context

[[Editorial note: This version was produced from the minutes, and then edited. It quite likely has some inaccuracies, so the authors would welcome corrections from those who attended the workshop. If you're a github person, https://github.com/sftcd/strint-report has this so you can contribute there.]]

The Vancouver IETF plenary concluded [vancouverplenary] that Pervasive Monitoring (PM) represents an attack on the Internet, and the IETF has begun to carry out the more obvious actions required to try to handle this attack. However, there are additional much more complex questions arising that need further consideration before any additional concrete plans can be made.

The W3C [1] and IAB [2] therefore decided to host a workshop [3] on the topic of "Strengthening the Internet Against Pervasive Monitoring" before IETF 89 [4] in London in March 2014. The FP7-funded STREWS [5] project organised the STRINT workshop on behalf of the IAB and W3C.

The main workshop goal was to discuss what can be done, especially by the two standards organisations IETF and W3C, against PM, both for existing Internet protocols (HTTP/1, SMTP, etc.) and for new ones (WebRTC, HTTP/2, etc.).

The starting point for the workshop was the existing IETF consensus that PM is an attack[I-D.farrell-perpass-attack].

2. Summary

The workshop was well attended (registration closed when the maximum capacity of 100 was reached, but more than 150 expressed a desire to register) and several people (about 165 at the maximum) listened to the streaming audio. The submitted papers (67 in total) were generally of good quality and all were published (see Appendix C), except for a few where authors who couldn't take part in the workshop preferred not to publish.

The chairs of the workshop summarised the workshop in the final session in the form of the following recommendations:

1. Well-implemented cryptography can be effective against PM and will benefit the Internet if used more, despite its cost, which is steadily decreasing anyway.

2. Traffic analysis also needs to be considered, but is less well understood in the Internet community: relevant research and protocol mitigations such as data minimisation need to be better understood.

3. Work should continue on progressing the PM threat model draft[I-D.barnes-pervasive-problem] discussed in the workshop.

4. Later, the IETF may be in a position to start to develop an update to BCP 72 [RFC3552], most likely as a new RFC enhancing that BCP and dealing with recommendations on how to mitigate PM and how to reflect that in IETF work.

5. The term "Opportunistic" has been widely used to refer to a possible mitigation strategy for PM. We need to document definition(s) for this term, as it is being used differently by different people and in different contexts. We may also be able to develop a cookbook-like set of related protocol techniques for developers. Since the workshop, the IETF's security area has taken up this work, most recently favouring the generic term "Opportunistic Security" (OS) [I-D.kent-opportunistic-security].

6. The technical community could do better in explaining the real technical downsides related to PM in terms that policy makers can understand.

7. Many User Interfaces (UI) could be better in terms of how they present security state, though this is a significantly hard problem. There may be benefits if certain dangerous choices were simply not offered anymore. But that could require significant co-ordination among competing software makers, otherwise some will be considered "broken" by users.

8. Ways to better integrate UI issues into the processes of IETF and W3C needs further discussion.

9. Examples of good software configurations that can be cut-and-paste'd for popular software, etc., can help. This is not necessarily standards work, but maybe the standards organisations can help and can work with those developing such package-specific documentation.

10. The IETF and W3C can do more so that default ("out-of-the-box") settings for protocols better protect security and privacy.

11. Captive portals, [6] (and some firewalls, too) can and should be distinguished from real man-in-the-middle attacks. This might mean establishing common conventions with makers of such

middleboxes, but might also need new protocols. However, the incentives for deploying such new middlebox features might not align.

3. Workshop goals

As stated, the STRINT workshop started from the position that PM is an attack. [I-D.farrell-perpass-attack] While some dissenting voices are expected and need to be heard, that was the baseline assumption for the workshop, and the high-level goal was to provide more consideration of that and how it ought to affect future work within the IETF and W3C.

At the next level down the goals of the STRINT workshop were to:

- Discuss and hopefully come to agreement among the participants on concepts in PM for both threats and mitigation, e.g., "opportunistic" as the term applies to cryptography.

- Discuss the PM threat model, and how that might be usefully documented for the IETF at least, e.g., via an update to BCP72. [7]

- Discuss and progress common understanding in the trade-offs between mitigating and suffering PM.

- Identify weak links in the chain of Web security architecture with respect to PM.

- Identify potential work items for the IETF, IAB, IRTF and W3C that help mitigate PM.

- Discuss the kinds of action outside the IETF/W3C context might help those done within the IETF/W3C.

4. Workshop structure

The workshop structure was designed to maximise discussion time. There were no direct presentations of submitted papers. Instead, the moderators of each session summarised topics that the Technical Programme Committee (TPC) had agreed based on the submitted papers. These summary presentations took at most 50% of the session and usually less.

Because the papers would not be presented during the workshop, participants were asked to read and discuss the papers beforehand, at

least those relevant to their fields of interest. (To help people choose papers to read, authors were asked to provide short abstracts.)

Most of the sessions had two moderators, one to lead the discussion, while the other managed the queue of people who wanted to speak. This worked well: everybody got a chance to speak and each session still ended on time.

The penultimate session consisted of break-outs. (Which turned out to be the most productive sessions of all, most likely simply due to the smaller numbers of people involved.) The subjects for the break-outs were agreed during the earlier sessions and just before the break-out session the participants collectively determined who would attend which.

5. Topics

The following sections contain summaries of the various sessions. See the minutes (see Appendix B) for more details.

5.1. Opening session

The first session discussed the goals of the workshop. Possible approaches to improving security in the light of pervasive monitoring include a critical look at what metadata is actually required, whether old (less secure) devices can be replaced with new ones, what are "low-hanging fruit" (issues that can be handled quickly and easily), and what level of security is "good enough": a good solution may be one that is good for 90% of people or 90% of organisations.

Some paricipants felt that standards are needed so that people can see if their systems conform to a certain level of security as well as easy to remember names for those standards, so that a buyer can immediately see that a product "conforms to the named intended standard."

5.2. Threats

One difference between "traditional" attacks and pervasive monitoring is modus-operandi of the attacker: typically, one determines what resources an attacker might want to target and at what cost and then one defends against that threat. But a pervasive attacker has no specific targets, other than to collect everything he can. The calculation of the cost of losing resources vs the cost of protecting them is thus different. And unlike someone motivated to make money, a PM attacker may not be concerned at the cost of the attack (or may

even prefer a higher cost, for "empire building" reasons).

The terminology used to talk about threats has to be chosen carefully (this was a common theme in several sessions), because we need to explain to people outside the technical community what they need to do or not do. For example, authentication of endpoints doesn't so much "protect against" man-in-the-middle (MITM) attacks as make them visible. The attacker can still attack, but it does not remain invisible while he does so. Somebody on either end of the conversation needs to react to the alert from the system: stop the conversation or find a different channel.

An interesting paradox is the role of big repositories of information, such as Facebook, Yahoo, Google, etc. Hopefully, they supervise their security better than the average Internet server, but they are also much more attractive as a target to attack. Avoiding overuse of such repositories for private or sensitive information may be a useful measure that increases the cost of collecting for a pervasive attacker. This is sometimes called the target-dispersal approach.

Lack of interoperability between systems is in itself a threat as it leads to work-arounds and compromises that may be less secure. And thus improving interoperability needs to be high on the list of priorities of standards makers and even more for implementers. Of course, testing, such as interop testing, is at some level, part of the process of IETF and W3C; and W3C is currently increasing its testing efforts.

5.3. Increase usage of security tools

The first session on Communication Security (COMSEC) tools looked at the question why existing security tools aren't used more.

The example of HTTPS is informative: it provides encryption and authentication and is widely available. In practice though, it is far from being used as much as it could be. It also has some problems. One problem is that certificate authorities (CA) are a potential weak link in the system. Any CA can issue a certificate for any server, and thus a single compromised CA can give a MITM the power to impersonate any server. Moreover, certificates can cost money, acquiring a certificate requires administrator time and effort, and certificates need to be replaced when they expire, which is not the normal case for web technologies, so many server administrators forget or don't bother, making the certificate infrastructure less relevant, and causing https to provide less security.

Some ideas were discussed for improving the CA system, e.g., via cross-certification of CAs and by means of "certificate transparency": a public, permanent log of who issued which certificate. [RFC6962]

Using other models than the hierarchical certificate model (as alternative or in combination) may also help. The PGP model, e.g., is a flat network where people verify the identity (public key) of people they meet. And then they trust, to a certain level, that those people verified the identity of other people. This works for certain types of communication (it was more deployed for e-mail). However, an identity only verified by a friend of a friend provides a lower level of trust.

Yet another model is "trust on first use" (TOFU). This is used quite effectively by SSH [RFC4252]. On the first connection, one has no way to verify that the received public key belongs to the server one is contacting, therefore, the key is accepted without further verification. But on the subsequent connections, one can verify that the received key is the same key as the first time. So a MITM has to be there on all connections, including the first, otherwise it will be detected by a key mismatch.

This works well for SSH, because people typically use SSH to communicate with a small number of servers over and over again. And, if they want, they may find a separate channel to get the public key (or its fingerprint). It may also work for Web servers used by small groups (the server of a sports club, a department of a company, etc.), but probably works less well for public servers that are visited once or a few times or for large services where many servers may be used.

A similar proposal [draft-ietf-websec-key-pinning] for an HTTP header introduces an aspect of TOFU into HTTP: Key pinning tells HTTP clients that for a certain time after receiving this certificate, they should not expect the certificate to change. If it does, even if the new certificate looks valid, the client should assume a security breach.

SIP [RFC3261] is a complex protocol, in part because it potentially needs several different intermediaries in different stages of the communication to deal with NAT traversal and to handle policy. SIP provides hop-by-hop encryption and end-to-end authentication in theory, but in practice many SIP providers disable these functions and interoperability for end-to-end security in SIP is perhaps not in a good state. The reasons for disabling end-to-end security here are understandable: to overcome lack of interoperability they often need to change protocol headers and modify protocol data. Some workshop

participants argued that SIP would never have taken off if it hadn't been possible for providers to monitor and interfere in communications in this way. Of course, that means an attacker can listen in just as easily.

A new protocol for peer-to-peer communication of video and audio (and potentially other data) is WebRTC. WebRTC re-uses many of the same architectural concepts as SIP, but there is a reasonable chance that it can do better in terms of protecting users: The people implementing the protocols and offering the service have different goals and interests. In particular, the first implementers are browser makers, who may have different business models from other more traditional Voice over IP providers.

XMPP suffers from yet another problem. It has encryption and authentication, and the OTR ("off the record") extension even provides what is called Perfect Forward Secrecy (PFS, compromising the current communication never gives an attacker enough information to decrypt past communications that he may have recorded). But, in practice, many people don't use XMPP at all, but rather Skype, WhatsApp or other instant-messaging tools with unknown or no security. The problem here seems to be one of user awareness. And though OTR does provide security, it is not well integrated with XMPP and nor is it available as a core feature of XMPP clients.

To increase usage of existing solutions, some tasks can be identified, though how those map to actions for e.g. IETF/W3C is not clear:

- Improvements to the certificate system, such as CT.
- Making it easier (cheaper, quicker) for system administrators to deploy secure solutions.
- Improve awareness of the risks. Identify which communities influence which decisions and what is the appropriate message for each.
- Provide an upgrade path that doesn't break existing systems or require that everybody upgrade at the same time. Opportunistic Security may be one model for that.

5.4. Policy issues and non-technical actions

Previous sessions already concluded that the problem isn't just technical, such as getting the right algorithms in the standards, fixing interoperability, or educating implementers and systems administrators. There are user interface issues and education issues

too. And there are also legal issues and policy issues for governments.

It appears that the public in general demand more privacy and security (e.g., for their children) but are also pessimistic about getting that. They trust that somebody assures that nothing bad happens to them, but they also expect to be spied on all the time.

(Perceived) threats of terrorism gave governments a reason to allow widespread surveillance, far beyond what may previously have been considered dangerous for freedom.

In this environment, the technical community will have a hard time developing and deploying technologies that fully counter PM. Which means there has to be action in the social and political spheres, too.

Technology isn't the only thing that can make life harder for attackers. Government-sponsored PM is indirectly affected by trade agreements and treaties and thus it makes sense to lobby for those to be as privacy-friendly as possible.

Court cases on the grounds of human rights can also influence policy, especially if they reach, for example, the European Court of Human Rights.

In medicine and law, it is common to have ethics committees, not so in software. Should standards bodies such as IETF and W3C have an ethics committee? While standards such as the Geolocation API [w3c-geo-api] have gotten scrutiny from privacy experts, but only in an ad-hoc manner. (W3C has permanent groups to review standards for accessibility and internationalisation. It also has a Privacy group, but that currently doesn't do the same kind of systematic reviews.)

As the Internet Draft draft-barnes-pervasive-problem-00 (included as paper 44 [8]) explains, PM doesn't just monitor the networks, but also attacks at the endpoints, turning organisations or people into (willing, unwilling, or unwitting) collaborators. One technical means of protection is thus to design protocols such that there are fewer potential collaborators, e.g., a provider of cloud storage cannot hand over plaintext for content that is encrypted with a key he doesn't have, and cannot hand over names if his client is anonymous.

It is important to distinguish between PM and fighting crime. PM is an attack, but a judge ordering the surveillance of a suspected criminal is not. The latter, often abbreviated in this context as LI (for Lawful Intercept), is outside the scope of this workshop.

5.5. Improving the tools

An earlier session discussed why existing COMSEC tools weren't used more. This second session on COMSEC therefore discussed what improvements and/or new tools were needed.

Discussion at the workshop indicated that an important meta-tool for improving existing security technology could be Opportunistic Security (OS). [I-D.kent-opportunistic-security]. The idea is that software is enhanced with a module that tries to encrypt communications when it detects that the other end also has the same capablility but otherwise leaves the communication continue in the old way. The detailed definition of OS is now being discussed by the IETF security area. [saag]

OS would protect against a passive eavesdropper but should also allow for endpoint authentication to protect against an active attacker (a MITM). As OS spreads, more and more communications would be encrypted (and hopefully authenticated) and thus there is less and less for an eavesdropper to collect.

Of course, an implementation of OS could give a false sense of security as well: some connections are encrypted, some are not. A user might see something like a padlock icon in browsers, but there was agreement at the workshop that such user interface features ought not be changed because OS is being used.

There is also the possibility that a MITM intercepts the reply from a server that says "yes, I can do encryption" and removes it, causing the client to fall back to an unencrypted protocol. Mitigations against this can be to have other channels of finding out a server's capabilities and remembering that a server could do encryption previously.

There is also, again, a terminology problem. The technical descriptions of OS talk about "silent fail" when a connection couldn't be encrypted and has to fall back to the old, unencrypted protocol. Actually, it's not a fail, it's no worse then it was before. A successful encryption would rather be a "silent improvement."

That raises the question of the UI: How do you explain to a user what their security options are, and, in case an error occurs, how do you explain the implications of the various responses?

The people working on encryption are mathematicians and engineers, and typically not the same people who know about UI. We need to involve the experts. We also need to distinguish between usability

of the UI, user understanding, and user experience. For an e-commerce site, e.g., it is not just important that the user's data is technically safe, but also that he feels secure. Otherwise he still won't buy anything.

When talking about users, we also need to distinguish the end user (who we typically think about when we talk about UI) from the server administrators and other technical people involved in enabling a connection. When something goes wrong (e.g., the user's software detects an invalid certificate), the message usually goes to the end user. But he isn't necessarily the person who can do something about it. E.g., if the problem is a certificate that expired yesterday, the options for the user are to break the connection (the safe choice, but it means he can't get his work done) or continue anyway (there could be a MITM...). The server administrator, on the other hand, could actually solve the problem.

Encryption and authentication have a cost, in terms of setting them up, but also in terms of the time it takes for software to do the calculations. The set-up cost can be reduced with sensible defaults, predefined profiles and cut-and-paste configurations. And for some connections, authentication without encryption could be enough, in the case that the data doesn't need to be kept secret, but it is important to know that it is the real data. Most mail UAs already provide independent options for encryption and signing, but Web servers only support authentication if the connection is also encrypted.

On the other hand, as e-mail also shows, it is difficult for users to understand what encryption and authentication do separately.

And it also has to be kept in mind that encrypting only the "sensitive" data and not the rest decreases the cost for an attacker, too: It becomes easy to know which connections are worth attacking. Selective field confidentiality is also more prone to lead to developer error, as not all developers will know the provenance of values to be processed.

One problem with the TOFU model as used by SSH (see explanation above) is that it lacks a solution for key continuity: When a key is changed (which can happen e.g., when a server is replaced or the software upgraded), there is no way to inform the client. (In practice, people use other means, such as calling people on the phone or asking their colleagues in the office, but that doesn't scale and doesn't always happen either.) An improvement in the SSH protocol could thus be a way to transfer a new key to a client in a safe way.

5.6. Hiding metadata

Encryption and authentication help protect the content of messages. Correctly implemented encryption is very hard to crack. (To get the content, an attacker would rather attempt to steal the keys, corrupt the encoding software, or get the content via a collaborator.) But encrypting the content doesn't hide the fact that you are communicating. This metadata (who talks to whom, when and for how long) is often as interesting as the content itself, and in some cases the size and timing of messages is even an accurate predictor of the content. So how to stop an attacker from collecting metadata, given that much of that data is actually needed by routers and other services to deliver the message to the right place?

It is useful to distinguish different kinds of metadata: explicit (or metadata proper) and implicit (sometimes called traffic data). Implicit metadata is things that can be derived from a message or are necessary for its delivery, such as the destination address, the size, the time, or the frequency with which messages pass. Explicit metadata is things like quality ratings, provenance or copyright data: data about the data, useful for an application, but not required to deliver the data to its endpoint.

A system like Tor hides much of the metadata by passing through several servers, encrypting all the data except that which a particular server needs to see. Each server thus knows which server a message came from and where he has to send it to, but cannot know where the previous server got it from or where the next server is instructed to send it. However, deliberately passing through multiple servers makes the communication slower than taking the most direct route and increases the amount of traffic the network as a whole has to process.

There are three kinds of measures that can be taken to make metadata harder to get: aggregation, contraflow and multipath (see paper 4 [9]). New protocols should be designed such that these measures are not inadvertently disallowed, e.g., because the design assumes that the whole of a conversation passes through the same route.

Aggregation means collecting conversations from multiple sources into one stream. E.g., if HTTP connections pass through a proxy, all the conversations appear to come from the proxy instead of from their original sources. (This assumes that telltale information in the headers is stripped by the proxy, or that the connection is encrypted.) It also works in the other direction: if multiple Web sites are hosted on the same server, an attacker cannot see which of those Web sites a user is reading. (This assumes that the name of the site is in the path info of the URL and not in the domain name,

otherwise watching DNS queries can still reveal the name.)

Contraflow means routing a conversation via one or more other servers than the normal route, e.g., by using a tunnel (e.g., with SSH or a VPN) to another server. Tor is an example of this. An attacker must watch more routes and do more effort to correlate conversations. (Again, this assumes that there is no telltale information left in the messages that leave the tunnel.)

Multipath splits up a single conversation (or a set of related conversations) and routes the parts in different ways. E.g., send a request via a satellite link and receive the response via a land line; or starting a conversation on a cellular link and continuing it via wifi. This again increases the cost for an attacker, who has to monitor and correlate multiple networks.

Protecting metadata automatically with technology at a lower layer than the application layer is difficult. The applications themselves need to pass less data, e.g., use anonymous temporary handles instead of permanent identifiers. There is often no real need for people to use the same identifier on different computers (smartphone, desktop, etc.) other than that the application they use was designed that way.

One thing that can be done relatively easily in the short term is going through existing protocols to check what data they send that isn't really necessary. One candidate mentioned for such a study was XMPP.

Fingerprinting is the process of distinguishing different senders of messages based on metadata: Clients can be recognised (or at least grouped) because their messages always have a combination of features that other clients do not have. Reducing redundant metadata and reducing the number of optional features in a protocol reduces the variation between clients and thus makes fingerprinting harder.

Traffic analysis is a research discipline that produces sometimes surprising findings, which are little known among protocol developers. Some collections of results are

- A selected bibliography on anonymity [10] by the Free Haven Project
- The yearly Symposium on Privacy Enhancing Technologies (PETS). [11]
- The yearly Workshop on Privacy in the Electronic Society (WPES). [12]

Techniques that deliberately change the timing or size of messages, such as padding, can also help reduce fingerprinting. Obviously, they make conversations slower and/or use more bandwidth, but in some cases that is not an issue, e.g., if the conversation is limited by the speed of a human user anyway. HTTP/2 has a built-in padding mechanism. However, it is not so easy to use these techniques well, and not actually make messages easier to recognise rather than harder.

Different users in different contexts may have different security needs, so maybe the priority can be a user choice. (If that can be done without making high-security users stand out from other users.) Although many people would not understand what their choices are, some do, such as political activists or journalists.

5.7. Deployment, intermediaries and middleboxes

Secure protocols have often been designed in the past for end-to-end security: Intermediaries cannot read or modify the messages. This is the model behind TLS for example.

But in practice people have more or less valid reasons to insist on intermediaries: companies filtering incoming and outgoing traffic for viruses or other reasons, giving priority to certain communications or caching to reduce bandwidth.

In the presence of end-to-end encryption and authentication, these intermediaries have two choices: using fake certificates to impersonate the endpoints or having access to the private keys of the endpoints. The former is a MITM attack that is difficult to distinguish from a more malicious one, the latter obviously decreases the security of the endpoints by copying supposedly protected data and concentrating such data in a single place.

As mentioned in Section 5.2 above, aggregation of data in a single place makes that place an attractive target. And in the case of PM even if the data is not concentrated physically in one place, but is under control of a single legal entity that can be made into a collaborator.

The way Web communication with TLS typically works is that the client authenticates the server, but the server does not authenticate the client at the TLS layer. (If the client needs to be identified, that is mainly done at the application layer via passwords or cookies.) Thus the presence of a MITM (middlebox) could be detected by the client (because of the incorrect certificate), but not by the server. If the client doesn't immediately close the connection, (which they do not in many cases), the server may thus disclose information that

the user would rather not have disclosed.

One widespread example of middleboxes is captive portals, as found on the wifi hotspots in hotels, airports, etc. Even the hotspots offering free access often intercept communications to redirect the user to a login or policy page.

When the communication they intercept is, e.g., the automatic update of your calendar program or a chat session, the redirect obviously doesn't work: these applications don't know how to display a Web page. With the increasing use of apps, it may be a while before the user actually opens a browser. The flood of error messages may also have as a result that the user no longer reads the errors, allowing an actual malicious attack to go unnoticed.

Some operating systems now come with heuristics that try to recognise captive portals and either automatically login or show their login page in a separate application. (But some hotspot providers apparently don't want automatic logins and actually reverse-engineered the heuristics to try and fool them.)

It seems some protocol is missing in this case. Captive portals shouldn't have to do MITM attacks to be noticed. Maybe something like an extension to DHCP that tells a connecting device about the login page can help, although that still doesn't solve the problem for devices that do not have a Web browser, such as game consoles or SIP phones. HTTP response code 511 (defined in [RFC6585]) is another attempt at a solution. (Partial, because it can only work at the moment the user uses a browser to connect to a Web site and doesn't use HTTPS.)

A practical problem with deployment of such a protocol may be that many such captive portals are very old and never updated. The hotel staff only knows how to reboot the system and as long as it works, the hotel has no incentive to buy a new one. As evidence of this: how many such systems require you to get a password and the ticket shows the price as zero? This is typically because the owner doesn't know how to reconfigure the hotspot, but he does know how to change the price in his cash register...

5.8. Break-out 1 - research

Despite some requests earlier in the workshop, the research break-out did not discuss clean-slate approaches. The challenge was rather that the relation between security research and standardisation needs improvement. Research on linkability is not yet well known in the IETF. But the other side of the coin needs improvement too: While doing protocol design, standardisation should indicate what specific

problems are in need of more research.

The break-out then made a non exclusive list of topics that are in need of further research:

- The interaction of compression and encryption as demonstrated by the CRIME SSL/TLS vulnerability [13]
- A more proactive deprecation of algorithms based on research results.
- Mitigation for return oriented programming attacks
- How to better obfuscate so called "metadata", how to make the existence of traffic and their endpoints stealthy

5.9. Break-out 2 - clients

Browsers are the first clients one thinks of when talking about encrypted connections, authentication and certificates, but there are many others.

Another common case of "false" alarms for MITM (after captive portals) is expired and mis-configured certificates. This is quite common in intranets, when the sysadmin hasn't bothered updating a certificate and rather tells his handful of users to just "click continue." The problem is on the one hand that users may not understand the difference between this case and the same error message when they connect to a server outside the company, and on the other hand that the incorrect certificate installed by the sysadmin is not easily distinguishable from an incorrect certificate from a MITM. The error message is almost the same and the user may just click continue again.

One way to get rid of such certificates is if client software no longer offers the option to continue after a certificate error. That requires that all major clients (such as browsers) change their behaviour at the same time, otherwise the first one to do so will be considered broken by users, because the others still work. Also it requires a period in which that software gives increasingly strong warnings about the cut-off date after which the connection will fail with this certificate.

Yet another source of error messages is self-signed certificates. Such certificates are actually only errors for sites that are not expected to have them. If a message about a self-signed certificate appears when connecting to Facebook or Google, you're clearly not connected to the real Facebook or Google. But for a personal Website

it shouldn't cause such scary warnings. There may be ways to improve the explanations in the error message and provide an easy way to verify the certificate (by e-mail, over the phone or some other channel) and import it.

5.10. Break-out 3 - on by default

One step in improving security is to require the relevant features, in particular encryption and authentication, to be implemented in compliant products: The features are labelled as MUST in the standard rather than MAY. This is sometimes referred to as MTI, Mandatory To Implement and is the current practice for IETF protocols. [RFC3365]

But that may not be enough to counter PM. It may be that the features are there, but not used, because only very knowledgeable users or sysadmins turn them on. Or it may be that implementations do not actually follow the MTI parts of specifications. Or it may be that some security features are implemented but interoperability for those doesn't really work. Or, even worse, it may be that protocol designers have only followed the letter of the MTI best practice and not its spirit, with the result that security features are hard to use or make deployment harder. One can thus argue that such features should be defined to be on by default.

Going further one might argue that these features should not even be options, i.e., there should be no way to turn them off. This is sometimes called MTU, Mandatory To Use.

The question raised at this session was for what protocols on-by-default is appropriate, and how to explain to the developers of such protocols that it is needed?

There would of course be resistance to MTU security from implemeters and deployments that practice deep packet inspection (DPI) and also perhaps from some governments. On the other hand, there may also be governments that outlaw protocols _without_ proper encryption.

This break-out concluded that there could be value in attempting to document a new Best Current Practice for the IETF that moves from the current MTI position to one where security features are on-by-default. Some of the workshop participants expressed interest in authoring a draft for such a new BCP and progressing that through the IETF consensus process. (Where it would no doubt be controversial.)

5.11. Break-out 4 - measurement

There was a small break-out on the idea of measurement as a way to encourage or gamify the increased use of security mechanisms. [[We

don't currently have notes from that.]]

5.12. Break-out 5 - opportunistic

This break out considered the use of the term "opportunistic" as it applies to crytpgraphic security and attempted to progress the work towards arriving at an agreed definition for use of that term, at it applies to IETF and W3C work.

While various terms had been used, with many people talking about opportunistic encryption, that usage was felt to be problematic both because it conflicted with the use of the same term in [RFC4322] and because it was being used differently in different parts of the community.

At the session it was felt that the term "opportunistic keying" was better, but as explained above subsequent list discussion resulted in a move to the term "Opportunistic Security" (OS).

Aside from terminology, disussion focused on the use of Diffie-Hellman (D-H) key exchange as the preferred mechanism of OS, with fall back to cleartext if D-H doesn't succeed as a counter for passive attacks.

There was also of course the desire to be able to easily escalate from countering passive attacks to also handling endpoint authentication and thereby also countering MITM attacks.

Making OS visible to users was again considered to be undesirable, as users could not be expected to distinguish between cleartext, OS and (one-sided or mutal) endpoint authentication.

Finally, it was noted that it may take some effort to establish how middleboxes might affect OS at different layers and that OS really is not suitable as the only migitation to use for high-sensitivity sessions such as financial transactions.

5.13. Unofficial Transport/Routing Break-out

Some routing and transport area directors felt a little left out by all the application layer break-outs:-) So they had their own brainstorm about what could be done at the Transport and Routing layers from which these notes resulted.

The LEDBAT [RFC6817] protocol was targeted towards a bulk-transfer service that is reordering and delay insensitive. Use of LDEBAT could offer the following benefits for an application:

a. Because it is reordering insensitive, traffic can be sprayed across a large number of forwarding paths. Assuming such different paths exist, this would make it more challenging to capture and analyze a full interaction.

b. The application can vary the paths by indicating per packet a different microflow. In IPv6, this can be done via different IPv6 flow labels. For IPv4, this can be done by encapsulating the IP packet into UDP and varying the UDP src port.

c. Since LEDBAT is delay insensitive and applications using it would need to be as well, it would be possible to obfuscate the application signatures by varying the packet lengths and frequency.

d. This can also hide the transport header (for IP in UDP).

e. If we could fix the reverse SPF check problem, perhaps the source could be hidden - but that has assumptions on trusted perimeters.

f. The use of LEDBAT is orthogonal to the use of encryption and provides different benefits (harder to intercept the whole conversation, ability to obfuscate the traffic analysis), and also has different costs (longer latency, new transport protocol usage) to its users.

We also discussed the idea of encrypting traffic from CE to CE as part of a L3VPN or such. This could allow hiding of addresses, including source, and headers. From my further conversation with Ron Bonica, some customers already do encryption (though not hiding the source address) like this. So, I'm not sure this is very practically useful as an enhancement except for encouraging deployment and use.

Finally, we discussed whether it would be useful to have a means of communicating where and what layers are doing encryption on an application's traffic path. The initial idea of augmenting ICMP has some issues (not visible to application, ICMP packets frequently filtered) as well as potential work (determining how to trust the report of encryption). It would be interesting to understand if such communication is actually needed and what the requirements would be.

6. After the workshop

Holding the workshop just before the IETF had the intended effect: a number of people went to both the workshop and the IETF. And they took the opportunity of being together at the IETF to continue the discussions.

Working groups of the IETF who met in London took the recommendations from the workshop into account. It was even the first item in the report about the IETF meeting by the IETF chair, Jari Arkko:

> _"Strengthening the security and privacy of the Internet continued to draw a lot of attention. The STRINT workshop organised by the IAB and W3C just before the IETF attracted 100 participants and over 60 papers. Even more people would have joined us, but there was no space. During the IETF meeting, we continued discussing the topic at various working groups. A while ago we created the first working group specifically aimed at addressing some of the issues surrounding pervasive monitoring. The Using TLS for Applications (UTA) working group had its first meeting in London. But many other working groups also address these issues in their own work. The TCPM working group discussed a proposal to add opportunistic keying mechanisms directly onto the TCP protocol. And the DNSE BOF considered the possibility of adding confidentiality support to DNS queries. Finally, there is an ongoing effort to review old specifications to search for areas that might benefit from taking privacy and data minimisation better into account."_ - [Arkko1]

Two papers that were written for the workshop, but not finished in time, are worth mentioning, too: One by the same Jari Arkko, titled "Privacy and Networking Functions" [Arkko2]; and one by Johan Pouwelse, "The Shadow Internet: liberation from Surveillance, Censorship and Servers" [draft-pouwelse-perpass-shadow-internet]

7. IANA considerations

There are none. We hope the RFC editor deletes this section.

8. Security considerations

This document does not define a technology but is all about security and privacy.

Plenary.
(C) Stonehouse Photographic [14]

9. Informative references

[Arkko1] Arkko, J., "IETF-89 Summary", March 2014, <http://www.ietf.org/blog/2014/03/ietf-89-summary/>.

[Arkko2] Arkko, J., "Privacy and Networking Functions", March 2014, <http://www.arkko.com/ietf/strint/draft-arkko-strint-networking-functions.txt>.

(Work in progress.)

[I-D.barnes-pervasive-problem]
Barnes, R., Schneier, B., Jennings, C., and T. Hardie, "Pervasive Attack: A Threat Model and Problem Statement", draft-barnes-pervasive-problem-00 (work in progress), January 2014.

[I-D.farrell-perpass-attack]
Farrell, S. and H. Tschofenig, "Pervasive Monitoring is an Attack", draft-farrell-perpass-attack-06 (work in progress), February 2014.

[I-D.kent-opportunistic-security]
Kent, S., "Opportunistic Security as a Countermeasure to Pervasive Monitoring", draft-kent-opportunistic-security-01 (work in progress), April 2014.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC3365] Schiller, J., "Strong Security Requirements for Internet Engineering Task Force Standard Protocols", BCP 61, RFC 3365, August 2002.

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.

[RFC4252] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", RFC 4252, January 2006.

[RFC4322] Richardson, M. and D. Redelmeier, "Opportunistic Encryption using the Internet Key Exchange (IKE)", RFC 4322, December 2005.

[RFC6585] Nottingham, M. and R. Fielding, "Additional HTTP Status Codes", RFC 6585, April 2012.

[RFC6817] Shalunov, S., Hazel, G., Iyengar, J., and M. Kuehlewind, "Low Extra Delay Background Transport (LEDBAT)", RFC 6817,

December 2012.

[RFC6962] Laurie, B., Langley, A., and E. Kasper, "Certificate Transparency", RFC 6962, June 2013.

[draft-ietf-websec-key-pinning]
Evans, C., Palmer, C., and R. Sleevi, "Public Key Pinning Extension for HTTP", February 2014.

(Work in progress.)

[draft-pouwelse-perpass-shadow-internet]
Pouwelse, J., Ed., "The Shadow Internet: liberation from Surveillance, Censorship and Servers", February 2014, <https://datatracker.ietf.org/doc/draft-pouwelse-perpass-shadow-internet/>.

(Work in progress.)

[saag] Area, S., "IETF Security Area mailing list", March 2014, <https://www.ietf.org/mail-archive/web/saag/current/maillist.html>.

[vancouverplenary]
IETF, "IETF 88 Technical Plenary Minutes", <http://www.ietf.org/proceedings/88/minutes/minutes-88-iab-techplenary>.

[w3c-geo-api]
Popescu, A., "Geolocation API Specification", October 2013, <http://www.w3.org/TR/geolocation-API/>.

[1] <http://www.w3.org/>

[2] <https://www.iab.org/>

[3] <https://www.w3.org/2014/strint/Overview.html>

[4] <https://www.ietf.org/meeting/89/index.html>

[5] <http://www.strews.eu/>

[6] <https://en.wikipedia.org/wiki/Captive_portal>

[7] <http://tools.ietf.org/html/bcp72>

[8] <https://www.w3.org/2014/strint/papers/44.pdf>

[9] <https://www.w3.org/2014/strint/papers/04.pdf>

[10] <http://freehaven.net/anonbib/>

[11] <http://www.informatik.uni-trier.de/~Ley/db/conf/pet/index.html>

[12] <http://www.informatik.uni-trier.de/~Ley/db/conf/wpes/index.html>

[13] <https://community.qualys.com/blogs/securitylabs/2012/09/14/crime -information-leakage-attack-against-ssltls>

[14] <http://www.stonehousephotographic.com/>

[15] <http://cordis.europa.eu/fp7/ict/>

[16] <http://blog.digital.telefonica.com/>

[17] <http://lists.i1b.org/pipermail/strint-attendees-i1b.org/>

[18] <https://twitter.com/search?q=%23strint>

[19] <http://www.w3.org/2014/02/28-strint-minutes.html>

[20] <http://down.dsg.cs.tcd.ie/strint-slides/s0-welcome.pdf>

[21] <http://down.dsg.cs.tcd.ie/strint-slides/s1-threat.pdf>

[22] <http://down.dsg.cs.tcd.ie/strint-slides/s2-comsec.pdf>

[23] <http://down.dsg.cs.tcd.ie/strint-slides/s3-policy.pdf>

[24] <http://www.w3.org/2014/03/01-strint-minutes.html>

[25] <http://down.dsg.cs.tcd.ie/strint-slides/s4-opportunistic.pdf>

[26] <http://down.dsg.cs.tcd.ie/strint-slides/s5-1metadata-pironti.pdf>

[27] <http://down.dsg.cs.tcd.ie/strint-slides/s5-2metadata-hardie.pdf>

[28] <http://down.dsg.cs.tcd.ie/strint-slides/s5-3metadata-cooper.pdf>

[29] <http://down.dsg.cs.tcd.ie/strint-slides/s6-deploy.pdf>

[30] <https://www.w3.org/2014/strint/slides/summary.pdf>

[31] <https://strint.pads.ccc.de/1>

[32] <https://www.w3.org/2014/strint/papers/01.pdf>

[33] <https://www.w3.org/2014/strint/papers/02.pdf>

[34] <https://www.w3.org/2014/strint/papers/03.pdf>

[35] <https://www.w3.org/2014/strint/papers/05.pdf>

[36] <https://www.w3.org/2014/strint/papers/06.pdf>

[37] <http://www.internetsociety.org/blog/tech-matters/2014/02/danger-new-internet-choke-points>

[38] <http://www.circleid.com/posts/20140218_mind_the_step_function_are_we_really_less_secure_than_a_year_ago/>

[39] <https://www.w3.org/2014/strint/papers/07.pdf>

[40] <https://www.w3.org/2014/strint/papers/08.pdf>

[41] <https://www.w3.org/2014/strint/papers/09.pdf>

[42] <https://www.w3.org/2014/strint/papers/10.pdf>

[43] <https://www.w3.org/2014/strint/papers/11.pdf>

[44] <https://www.w3.org/2014/strint/papers/12.pdf>

[45] <https://www.w3.org/2014/strint/papers/13.pdf>

[46] <https://www.w3.org/2014/strint/papers/14.pdf>

[47] <https://www.w3.org/2014/strint/papers/15.pdf>

[48] <https://www.w3.org/2014/strint/papers/17.pdf>

[49] <https://www.w3.org/2014/strint/papers/19.pdf>

[50] <https://www.w3.org/2014/strint/papers/20.pdf>

[51] <https://www.w3.org/2014/strint/papers/21.pdf>

[52] <https://www.w3.org/2014/strint/papers/22.pdf>

[53] <https://www.w3.org/2014/strint/papers/23.pdf>

[54] <https://www.w3.org/2014/strint/papers/24.pdf>

[55] <https://www.w3.org/2014/strint/papers/25.pdf>

[56] <https://www.w3.org/2014/strint/papers/26.pdf>

[57] <https://www.w3.org/2014/strint/papers/27.pdf>

[58] <https://www.w3.org/2014/strint/papers/28.pdf>

[59] <https://www.w3.org/2014/strint/papers/30.pdf>

[60] <https://www.w3.org/2014/strint/papers/31.pdf>

[61] <https://www.w3.org/2014/strint/papers/32.pdf>

[62] <https://www.w3.org/2014/strint/papers/33.pdf>

[63] <https://www.w3.org/2014/strint/papers/34.pdf>

[64] <https://www.w3.org/2014/strint/papers/35.pdf>

[65] <https://www.w3.org/2014/strint/papers/36.pdf>

[66] <https://www.w3.org/2014/strint/papers/37.pdf>

[67] <https://www.w3.org/2014/strint/papers/38.pdf>

[68] <https://www.w3.org/2014/strint/papers/39.pdf>

[69] <https://www.w3.org/2014/strint/papers/40.pdf>

[70] <https://www.w3.org/2014/strint/papers/41.pdf>

[71] <https://www.w3.org/2014/strint/papers/42.pdf>

[72] <https://www.w3.org/2014/strint/papers/43.pdf>

[73] <https://www.w3.org/2014/strint/papers/45.pdf>

[74] <https://www.w3.org/2014/strint/papers/46.pdf>

[75] <https://www.w3.org/2014/strint/papers/47.pdf>

[76] <https://www.w3.org/2014/strint/papers/48.pdf>

[77] <https://www.w3.org/2014/strint/papers/49.pdf>

[78] <https://www.w3.org/2014/strint/papers/50.pdf>

[79] <https://www.w3.org/2014/strint/papers/51.pdf>

[80] <https://www.w3.org/2014/strint/papers/52.pdf>

[81] <https://www.w3.org/2014/strint/papers/53.pdf>

[82] <https://www.w3.org/2014/strint/papers/54.pdf>

[83] <https://www.w3.org/2014/strint/papers/55.pdf>

[84] <https://www.w3.org/2014/strint/papers/56.pdf>

[85] <https://www.w3.org/2014/strint/papers/57.pdf>

[86] <https://www.w3.org/2014/strint/papers/58.pdf>

[87] <https://www.w3.org/2014/strint/papers/59.pdf>

[88] <https://www.w3.org/2014/strint/papers/60.pdf>

[89] <https://www.w3.org/2014/strint/papers/61.pdf>

[90] <https://www.w3.org/2014/strint/papers/62.pdf>

[91] <https://www.w3.org/2014/strint/papers/63.pdf>

[92] <https://www.w3.org/2014/strint/papers/64.pdf>

[93] <https://www.w3.org/2014/strint/papers/65.pdf>

[94] <https://www.w3.org/2014/strint/papers/66.pdf>

[95] <https://www.cs.tcd.ie/Stephen.Farrell/>

[96] <http://www.w3.org/People/Rigo/>

[97] <http://www.tschofenig.priv.at/wp/?page_id=5>

Appendix A. Logistics

The workshop was organised by the STREWS [5] project (a research project funded under the European Union's 7th Framework Programme [15]), as the first of two workshops in its work plan. The

organisers were supported by the IAB and W3C, and, for the local organisation, by Telefonica Digital. [16]

One of the suggestions in the project description of the STREWS project was to attach the first workshop to an IETF meeting. The best opportunity was IETF 89 [4] in London, which would begin on Sunday March 2, 2014. Telefonica Digital offered meeting rooms at its offices in central London for the preceding Friday and Saturday, just minutes away from the IETF's location.

The room held 100 people, which was thought to be sufficient. There turned out to be more interest than expected and we could have filled a larger room, but 100 people is probably an upper limit for good discussions anyway.

Apart from the usual equipment in the room (projector, white boards, microphones, coffee...), we also set up some extra communication channels:

- A mailing list where participants could discuss the agenda and the published papers about three weeks in advance of the workshop itself. (Only participants were allowed to write to the mailing list, but the archive [17] is public.)

- Publicly advertised streaming audio (one-way only). At some point, no less than 165 people were listening.

- An IRC channel for live minute taking, passing links and other information, and as a help for remote participants to follow the proceedings.

- An Etherpad, where the authors of papers could provide an abstract of their submissions, to help participants who could not read all 66 papers in full in advance of the workshop. (The abstracts were also used on the workshop's Web site and are reproduced in this report (Appendix C).)

- A "Twitter hashtag" (#strint). Four weeks after the workshop, there were still a few new messages [18] about events related to workshop topics.

Appendix B. Agenda

This was the final agenda of the workshop, as determined by the TPC and participants on the mailing list prior to the workshop. The included links are to the slides that the moderators used to introduce each discussion topic and to the minutes.

B.1. Friday 28 February

(minutes [19])

13:00 Registration, Coffee, play with n/w, power, find seat

14:00 Workshop starts, welcome, logistics, opening/overview [slides] [20]

* Goal is to plan how we respond to PM threats
* Specific questions to be discussed in sessions
* Outcomes are actions for IETF, W3C, IRTF, etc.

14:30 I. Threats - What problem are we trying to solve? (Presenter: Richard Barnes; Moderator: Cullen Jennings) [slides] [21]

* What attacks have been described? (Attack taxonomy)
* What should we assume the attackers' capabilities are?
* When is it really "pervasive monitoring" and when is it not?
* Scoping - what's in and what's out? (for IETF/W3C)

15:30 Break

16:00 II. COMSEC 1 - How can we increase usage of current COMSEC tools? (Presenter: Hannes Tschofenig; Moderator: Leif Johansson) [slides] [22]

* Whirlwind catalog of current tools
* Why aren't people using them? In what situations are / aren't they used?
* Securing AAA and management protocols - why not?
* How can we (IETF/W3C/community) encourage more/better use?

17:30 Break

17:45 III. Policy - What policy / legal/ other issues need to be taken into account? (Presenter: Christine Runnegar; Moderator: Rigo Wenning) [slides] [23]

* What non-technical activities do we need to be aware of?

* How might such non-technical activities impact on IETF/W3C?

* How might IETF/W3C activities impact on those non-technical activities?

18:30 Session IV - Saturday plan, open-mic, wrap up day

19:00 Social event

Break out.
(C) Stonehouse Photographic [14]

B.2. Saturday 1 March

(minutes [24])

09:00 Welcome again, logistics

09:15 IV. COMSEC 2 - What improvements to COMSEC tools are needed?(Presenter: Mark Nottingham; Moderator: Steve Bellovin) [slides] [25]

* Opportunistic encryption - what is it and where it might apply

* Mitigations aiming to block PM vs. detect PM - when to try which?

10:30 Break

10:45 V. Metadata - How can we reduce the metadata that protocols expose? (Presenter: Alfredo Pironti [slides] [26] / Ted Hardie [slides] [27]; Moderator: Alissa Cooper [slides] [28])

* Meta-data, fingerprinting, minimisation

* What's out there?

* How can we do better?

12:00 Lunch (Buffet)

13:00 VI. Deployment - How can we address PM in deployment / operations? (Presenter: Eliot Lear; Moderator: Barry Leiba) [slides] [29]

* "Mega"-commercial services (clouds, large scale email & SN, SIP, WebRTC...)

* Target dispersal - good goal or wishful thinking?

* Middleboxes: when a help and when a hindrance?

14:30 Break

15:00 VII. 3 x Break-out Sessions / Bar-Camp style (Hannes Tschofenig)

* Content to be defined during meeting, as topics come up

* Sum up at the end to gather conclusions for report

15:00 Break-outs:

1. Research Questions (Moderator: Kenny Paterson)

    + Do we need more/different crypto tools?

    + How can applications make better use of COMSEC tools?

    + What research topics could be handled in IRTF?

    + What other research would help?

2. clients

3. on by default

4. measuring

5. opportunistic

16:15 VIII. Break-out reports, Open mic & Conclusions - What are we going to do to address PM? [slides] [30]

* Gather conclusions / recommendations / goals from earlier sessions

17:15 End

Whiteboard notes.
(C) Stonehouse Photographic [14]

Appendix C. The submitted papers

[[sort-papers: Group the papers by (rough) topic? --Bert]]

The following papers were submitted to the workshop. The abstracts were provided by the authors themselves. (We set up an editable page ("Etherpad") [31] where the authors could insert them.)

C.1. Privacy Protected Email - Phillip Hallam-Baker

01.pdf [32] - This proposal is two things: First it shows that with some small adjustments to S/MIME and PGP we can merge two competing end-to-end security proposals that are too hard for people to use into one scheme that provides a useful degree of security with no thought from the user. In cases where the user has security concerns they can easily determine that they are met. The second part of the proposal is that it the Trust set deployed to secure email encryption can be leveraged to solve pretty much every other end-to-end security requirement. If people generate keys for their email we can secure chat, video, 2-factor authentication as well.

C.2. Opportunistic Encryption for MPLS - Stephen Farrell, Adrian Farrrell

02.pdf [33] - This is an early proposal for a way to do open-channel D-H key agreement and encryption in MPLS. Two things are maybe interesting: a) its an example of trying to add confidentiality to an existing protocol with making PM harder as a specific goal and b) maybe it shows that there could be a benefit in a generic protocol for after-the-fact MITM detection for such cases. It'd probably be most interesting to discuss (a) as one example of something we want to do more generally and not the specifics of MPLS at the workshop; and I'd be interested in whether or not (b) is tractable (I'm not sure).

C.3. Overcoming the Friend-or-Foe Paradigm in Secure Communication - Sebastian Gajek, Jan Seedorf, Marc Fischlin, Oezguer Dagdalen

03.pdf [34] - Essentially, our point is that with the existing end-to-end client-server security paradigm, e.g. as instantiated in TLS, the "good guys" often actually have to mount attacks in order for middleboxes (which are on the path between client ans server being able) to perform their job. The good guys are thus technically indistinguishable from the bad guys.

Concretely, we are proposing to extend TLS in a way that would allow authorized modification of certain, dedicated parts of the TLS payload by middleboxes, while still allowing for integrity verification by clients. The crypto for such "Interferable Secure Communication" exists and we think it is feasible to extend TLS in this way in a reasonable timeframe.

C.4. Flows and Pervasive Monitoring - Ted Hardie

04.pdf [9] - This document describes methods that may hinder a pervasive monitor's efforts to derive metadata from flows. There are three main methods discussed in the paper: aggregation, contraflow, and multipath. These are largely side-effects of other efforts at this time, but the paper discusses how they might fit into the design space of efforts intended to combat pervasive monitoring and the related consequences for network operations.

C.5. BetterCrypto.org Applied Crypto Hardening - Aaron Zauner, L. Aaron Kaplan

05.pdf [35] - BetterCrypto is a community-driven project where admins, engineers, cryptographers, security researchers alike participate in finding well researched best-practices for commonly deployed networked applications and infrastructure. We try to outline a proper interim solution until better protocols and standards are widely deployed. Our hope is that we can contribute to a safer internet for all and better understanding of cryptographic primitives for the operations community that needs to deploy sound security on the public internet. Our focus group: sysadmins / ops.

C.6. A Complimentary Analysis (The Danger Of The New Internet Choke Points) - Andrei Robachevsky, Christine Runnegar, Karen O'Donoghue, Mat Ford

06.pdf [36] - The ongoing disclosures of pervasive surveillance of Internet users' communications and data by national signals intelligence agencies have prompted protocol designers, software and hardware vendors, as well as Internet service and content providers, to re-evaluate prevailing security and privacy threat models and to refocus on providing more effective security and confidentiality. At IETF88, there was consensus to address pervasive monitoring as an attack and to consider the pervasive attack threat model when designing a protocol. In this paper, we offer a complimentary analysis. We identify some of the components of the Internet architecture that provide attractive opportunities for wholesale monitoring and/or interception, and, therefore, represent architectural vulnerabilities, or choke points. We also suggest possible mitigation strategies and pose some of the questions that

need to be considered if the Internet is to evolve to reduce such vulnerabilities. Finally, we identify some significant areas of tension or trade-offs, and we consider possible areas for additional efforts. Also: danger-new-internet-choke-points [37] and mind_the_step_function [38]

C.7. Trust Issues with Opportunistic Encryption - Scott Rose, Stephen Nightingale, Doug Montgomery

07.pdf [39] - "Once is happenstance. Twice is coincidence. Three times is enemy action"

The lack of authentication in opportunistic encryption could have the perverse affect of putting more end users at risk: thinking that they are "secure", an end user may divulge private information to an imposter instead of the service they believe they have contacted. When adding protection mechanisms to protocols, designers and implementers should not downplay the importance of authentication in order to make opportunistic encryption easier to deploy. We advocate that while opportunistic encryption can solve one set of problems, authentication is often desired by end users.

C.8. Challenges with End-to-End Email Encryption - Jiangshan Yu, Vincent Cheval, Mark Ryan

08.pdf [40] - In this paper we show how the use of an extended certificate transparency can build a secure end-to-end email or messaging system using PKI without requiring trusted parties nor complex p2p key-signing arrangements such as PGP. This makes end-to-end encrypted mail possible, and users do not need to understand or concern themselves with keys or certificates. In addition, we briefly present some related concerns i.e. metadata protection, key loss mitigation, spam detection, and the security of webmail.

C.9. Strengthening the path and strengthening the end-points - Xavier Marjou, Emile Stephan, Jean-Michel Combes, Iuniana Oprescu

09.pdf [41] - Internet data is more and more subject to pervasive monitoring. This paper investigates ways of enhancing this situation depending on where such pervasive monitoring may occur. There are two different locations to secure: the endpoints and the path between these endpoints. In the present document, we also emphasize the fact that encryption, although bringing additional data confidentiality, might in some cases contradict security's two other pillars, which are availability and integrity.

C.10. SIP is Difficult - Jon Peterson

10.pdf [42] - While SIP is widely used as a protocol for real-time communications, it is very difficult to secure from pervasive monitoring. In fact, one could argue that SIP's susceptibility to mass surveillance was essential to its success in the marketplace. This paper shows why SIP's design left the door open for eavesdropping, and what lessons RTCWeb could learn from this.

C.11. Thoughts of Strengthening Network Devices in the Face of Pervasive Surveillance - Dacheng Zhang, Fuyou Miao

11.pdf [43] - The material released by Edward Snowden has raised serious concerns about pervasive surveillance. People worry that their privacy is not properly protected when they are using the Internet. Network product vendors also encounter the doubts on the security of their products (e.g., routers, switches, firewalls). Such doubts are seriously damaging the Internet ecosystem. In this paper we try to analyze the affects brought by the Snowden scandal on our ability to trust products at the core of the Internet and discuss what the standard organization can do to help vendors address these security concerns.

C.12. Opportunistic Encryption for HTTP URIs - Mark Nottingham

12.pdf [44] - This is a proposed method for using TLS with http:// URIs under discussion in the HTTPbis WG, in particular for HTTP/2 but also applicable to HTTP/1. One of the biggest decisions to make is whether or not to require the certs to validate in this scenario.

C.13. Cyberdefense-Oriented Multilayer Threat Analysis - Yuji Sekiya, Daisuke Miyamoto, Hajime Tazaki

13.pdf [45]

C.14. A Threat Model for Pervasive Passive Surveillance - Brian Trammell, Daniel Borkmann, Christian Huitema

14.pdf [46] - This document elaborates a threat model for pervasive surveillance, assuming an adversary with an interest in indiscriminate eavesdropping that can passively observe network traffic at every layer at every point in the network between the endpoints. We provide guidelines on evaluating the observability and inferability of information and metainformation radiated from Internet protocols. The central message to protocol designers: pervasive encryption for confidentiality, protocol and implementation design for simplicity and auditability, flexibility to allow fingerprinting resistance, and moving away from static identifiers

can increase protocol-level resistance to pervasive surveillance.

C.15. Why Provable Transparency is Useful Against Surveillance - Ben Laurie

15.pdf [47]

C.16. Withheld

C.17. Monitoring message size to break privacy - Current issues and proposed solutions - Alfredo Pironti

17.pdf [48] - One of the Internet traffic features that can be easily collected by passive pervasive monitoring is the size of the exchanged messages, or the total bandwidth used by a conversation. Several works have showed that careful analysis of this data can break users' expected privacy, even for encrypted traffic. Despite this, little has been done in practice to hide message sizes, perhaps because deemed too inefficient or not a realistic threat.

In this short paper, we contextualize message size analysis in the wider pervasive monitoring scenario, which encompasses other powerful analysis techniques, and we re-state the severity of the privacy breach that message size analysis constitutes. We finally discuss proposals to fix this issue, considering practical aspects such as required developer awareness, ease of deployment, efficiency, and interaction with other countermeasures.

C.18. Withheld

C.19. Making The Internet Secure By Default - Michael H. Behringer, Max Pritkin, Steinthor Bjarnason

19.pdf [49] - Pervasive monitoring on the Internet is enabled by the lack of general, fundamental security. In his presentation at the 88th IETF Bruce Schneier called for ubiquitous use of security technologies to make pervasive monitoring too expensive and thus impractical. However, today security is too operationally expensive, and thus only used where strictly required. In this position paper we argue that all network transactions can be secure by default, with minimal or no operator involvement. This requires an autonomic approach where all devices in a domain enrol automatically in a trust domain. Once they share a common trust anchor they can secure communications between themselves, following a domain policy which is by default secure. The focus of this proposal is the network itself, with all protocols between network elements, including control plane protocols (e.g., routing protocols) and management plane protocols (e.g., SSH, netconf, etc). The proposal is evolutionary and allows a

smooth migration from today's Internet technology, device by device.

C.20. Increasing HTTP Transport Confidentiality with TLS Based Alternate Services - Patrick McManus

20.pdf [50]

C.21. Balance - Societal security versus individual liberty - Scott Cadzow

21.pdf [51]

C.22. Strengthening the Extensible Messaging and Presence Protocol (XMPP) - Peter Saint-Andre

22.pdf [52] - This document describes existing and potential future efforts at strengthening the Extensible Messaging and Presence Protocol (XMPP), for discussion at the W3C/IAB workshop on Strengthening the Internet Against Pervasive Monitoring (STRINT).

C.23. The Internet We Want or the Internet We Deserve? - David Rogers

23.pdf [53]

C.24. Beyond Encrypt Everything: Passive Monitoring - Mark Donnelly, Sam Hartman

24.pdf [54]

C.25. Examining Proxies to Mitigate Pervasive Surveillance - Eliot Lear, Barbara Fraser

25.pdf [55] - The notion of pervasive surveillance assumes that it is possible for an attacker to have access to all links and devices between end points, as well as end points themselves. We examine this threat is some detail with an eye toward whether trusted intermediaries can provide relief from the attack. We go on to examine the costs associated with the various remediation methods. In at least one case, we challenge the notion that one should encrypt absolutely everything in all cases, as was implied in at least one threat analysis. Finally we summarize in a set of four principles that should be considered in future work.

C.26. Spontaneous Wireless Networking to Counter Pervasive Monitoring - Emmanuel Baccelli, Oliver Hahm, Matthias Waehlisch

26.pdf [56] - Several approaches can be employed to counter pervasive monitoring at large scale on the Internet. One category of

approaches aims to harden the current Internet architecture and to increase the security of high profile targets (data centers, exchange points etc.). Another category of approaches aims instead for target dispersal, i.e. disabling systematic mass surveillance via the elimination of existing vantage points, thus forcing surveillance efforts to be more specific and personalized. This paper argues how networking approaches that do not rely on central entities - but rather on spontaneous interaction, as locally as possible, between autonomous peer entities - can help realize target dispersal and thus counter pervasive monitoring.

C.27. Is Opportunistic Encryption the Answer? Practical Benefits and Disadvantages - John Mattsson

27.pdf [57] - In this paper, we give an overview of various opportunistic and unauthenticated encryption techniques, and discuss their benefits, limits, and disadvantages. We recommend the Internet community to clearly define the term "opportunistic encryption" or to use other terms. We conclude that while opportunistic and unauthenticated encryption certainly has its uses and may with the right choices provide good enough security for a low cost, general deployment of unauthenticated encryption is not an effective way to thwart pervasive monitoring.

C.28. Clearing off the Cloud over the Internet of Things - Carsten Bormann, Stefanie Gerdes, Olaf Bergmann

28.pdf [58] - As was foreshadowed by product introductions in 2013, the Consumer Electronics Show 2014 has seen the introduction of a large number of "Internet of Things" (IoT) innovations. Almost all of these have in common that they are meant to operate via Cloud-based services. In the light of the recent attention to threats by state-level tenacious attackers with significant infrastructure (STASI), in particular to their practice of pervasive monitoring, we discuss the implications of a cloud-centric IoT landscape, and attempt to outline a set of principles as a program to improve the long-term outlook.

C.29. Withheld

C.30. The Trust-to-Trust Model of Cloud Services - Alissa Cooper, Cullen Jennings

30.pdf [59]

C.31. Linkability Considered Harmful - Leif Johansson

31.pdf [60] - Current debate on pervasive monitoring often focus on passive attacks on the protocol and transport layers but even if these issues were eliminated through the judicious use of encryption, roughly the same information would still be available to an attacker who is able to (legally or otherwize) obtain access to linked data sets which are being maintained by large content and service providers.

C.32. Simple Opportunistic Encryption - Andrea Bittau, Michael Hamburg, Mark Handley, David Mazieres, Dan Boneh

32.pdf [61] - Network traffic encryption is becoming a requirement, not an option. Enabling encryption will be a communal effort so a solution that gives partial benefits until fully deployed is needed. A solution that requires little changes to existing infrastructure will also help as it can be quickly deployed to give immediate short-term benefits. We argue that tcpcrypt, a TCP option for opportunistic encryption is the path of least-resistance for a solution against large-scale traffic encryption. Tcpcrypt requires no changes to applications, is compatible with existing networks (works with NATs), and just works by default. It is high performance, so it can be deployed on servers without much concern. tcpcrypt attempts to maximize security for any given setting. By default, it will protect against passive eavesdropping, and also allows detecting large scale interception. With authentication, tcpcrypt can provide full security against active attackers and so it is a complete solution both for the short-term and long-term.

C.33. An Architecture for a Secure Cloud Collaboration System - Cullen Jennings, Suhas Nandakumar

33.pdf [62] - The Internet technical community is looking at ways to address pervasive attacks as described in several other internet drafts. [I-D.barnes-pervasive-problem] describes threat model to characterize various pervasive attacks on the Internet communications. There are many systems that need to be secured against such attacks but this paper considers one possible way to secure cloud based collaborations systems. At a high level, this paper sugests that users or enterprises could run a key server that manages the keys to access their content. The cloud service provider would not have access to decrypt the data stored in the cloud but various users of the cloud service could get the keys to encrypt and decrypt the contents of collaboration sessions facilitated by the cloud service. This does not protect the meta data of who is talking to who but can help protect the content of the conversations.

C.34. Security and Simplicity - Steven Bellovin

34.pdf [63]

C.35. Privacy at the Link Layer - Piers O'Hanlon, Joss Wright, Ian Brown

35.pdf [64] - This paper gives an overview of the privacy issues around the use of link layer identifiers and associated protocols. Whilst the IETF generally specifies IP level protocols it does also address the link layer in protocols such as address resolution, network attachment detection, tunnelling and router redundancy.

The indiscriminate broadcast of a device's MAC address, a unique and effectively personal identifier, allows for unregulated and broad-scale tracking of individuals via their personal devices, whether or not those devices have made use of a particular service or not. These addresses typically remain unchanged for the lifetime of a device, creating a persistent, lifelong tracking capability. The collation of such addresses, primarily WiFi and Bluetooth, has been been gathering pace and is already in use by organisations such as security agencies and advertisers.

Ephemeral addresses are used further up the stack so why not at the link layer? As default devices should use a randomised MAC address and any higher level associations can be maintained as and when approved by the user. Moreover various other 'performance enhancing' approaches further degrade the privacy of individuals such as proactive discovery of WLAN SSIDs, Detection of Network Attachment (DNA), Wireless ISP roaming (WISPr), name lookups and so on.

All these mechanisms need to be re-examined in the light of pervasive monitoring.

C.36. Erosion of the moral authority of middleboxes - Joe Hildebrand

36.pdf [65] - Many middleboxes on the Internet attempt to add value to the connections that traverse that point on the network. Problems in their implementations erode the moral authority that otherwise might accrue to the legitimate value that they add.

C.37. Policy Responses, Implications and Opportunities - Joseph Lorenzo Hall & Runa Sandvik

37.pdf [66] - We raise issues for discussion that lie in the interface between policy and technology. Specifically, we discuss 1) routing, processing and data localization policy mandates (i.e., new laws that may affect how data flows through the 'net; 2) the

uncertain possibility of dilution of credibility of IETF and w3c given what we've seen with NIST after NSA-coziness allegations; 3) the claim that strenghtening the internet and web will "help the bad guys" and the dubious need for "lawful intercept" funcationality; and 3) abusive content, cryptography as a controlled export technology, and the need to standardize more anonymity primitives (onion routing, pluggable transport protocols). We also highlight our own work in ensuring that technologists have a voice in policy environments and discuss a few interventions we coordinated over the past year, focusing on software backdoors and NSA surviellance.

C.38. Is it time to bring back the hosts file? - Peter Eckersley

38.pdf [67]

C.39. Metaphors matter; application-layer; distribute more - Larry Masinter

39.pdf [68] -

1. Dont say Attack: IETF should stay away from political theatre: changing protocols or workflows not because the change works but just to say you did something. Metaphors matter.

2. For most relevant threats, traffic analysis is enough, and encyption doesnt mitigate.

3. The only deployable protection - if that is what is wanted - means shifting architecture from client-server to mesh.

C.40. Levels of Opportunistic Privacy Protection for Messaging-Oriented Architectures - Dave Crocker, Pete Resnick

40.pdf [69] - Messaging protection against pervasive monitoring (PM) needs to cover primary payload, descriptive meta-data, and traffic-related analysis. Complete protection against PM, for traffic through complex handling sequences, has not yet been achieved reliably in real-world operation. Consequently, this document considers a range of end-to-end, object-based mechanisms, distinct from channel-based mechanisms. Each approach offers incremental protection levels that can be provided with existing, or low-risk, component technologies, such as through the DNS and MIME conventions.

C.41. Fingerprinting Guidance for Web Specification Authors - Nick Doty

41.pdf [70] - http://w3c.github.io/fingerprinting-guidance/ - Exposure of settings and characteristics of browsers can impact user privacy by allowing for browser fingerprinting. This document

defines different types of fingerprinting, considers distinct levels of mitigation for the related privacy risks and provides guidance for Web specification authors on how to balance these concerns when designing new Web features.

C.42. Eradicating Bearer Tokens for Session Management - Philippe De Ryck, Lieven Desmet, Frank Piessens, Wouter Joosen

42.pdf [71] - Session management is a crucial component in every modern web application. It links multiple requests and temporary stateful information together, enabling a rich and interactive user experience. The de facto cookie-based session management mechanism is however flawed by design, enabling the theft of the session cookie through simple eavesdropping or script injection attacks. Possession of the session cookie gives an adversary full control the user's sover ession, allowing him to impersonate the user to the target application and perform transactions in the user's name. While several alternatives for secure session management exist, they fail to be adopted due to the introduction of additional roundtrips and overhead, as well as incompatibility with current Web technologies, such as third-party authentication providers, or widely deployed middleboxes, such as web caches. We identify four key objectives for a secure session management mechanism, aiming to be compatible with the current and future Web. We propose SecSess, a lightweight session management mechanism based on a shared secret between client and server, used to authenticate each request. SecSess ensures that a session remains under control of the parties that established it, and only introduces limited overhead. During session establishment, SecSess introduces no additional roundtrips and only adds 4.3 milliseconds to client-side and server-side processing. Once a session is established, the overhead becomes negligible (<0.1ms), and the average size of the request headers is even smaller than with common session cookies. Additionally, SecSess works well with currently deployed systems, such as web caches and third-party services. SecSess also supports a gradual migration path, while remaining compatible with currently existing applications.

C.43. STREWS Web-platform security guide: security assessment of the Web ecosystem - Martin Johns, Lieven Desmet

43.pdf [72] - In this document, we report on the Web-platform security guide, which has been developed within the EC-FP7 project STREWS. Based on their research, the STREWS consortium argues that in order to strengthening the Internet (e.g. against pervasive monitoring), it is crucial to also strengthen the web application ecosystem, the de-facto Internet application platform.

C.44. Pervasive Attack: A Threat Model and Problem Statement - Richard Barnes, Bruce Schneier, Cullen Jennings, Ted Hardie

44.pdf [8] - Documents published in 2013 have revealed several classes of "pervasive" attack on Internet communications. In this document, we review the main attacks that have been published, and develop a threat model that describes these pervasive attacks. Based on this threat model, we discuss the techniques that can be employed in Internet protocol design to increase the protocols robustness to pervasive attacks.

C.45. Cryptech - Building a More Assured HSM with a More Assured Tool-Chain - Randy Bush

45.pdf [73]

C.46. Replacing passwords on the Internet AKA post-Snowden Opportunistic Encryption - Ben Laurie, Ian Goldberg

46.pdf [74]

C.47. End-User Concerns about Pervasive Internet Monitoring: Principles and Practice - Tara Whalen, Stuart Cheshire, David Singer

47.pdf [75] - This position paper will discuss pervasive monitoring on the Internet from the perspective of end users: what are overarching concerns around pervasive monitoring, and what are some steps that could be taken to address those concerns? We begin by exploring a preliminary set of characteristics of systemic surveillance, which can be used to pinpoint dominant concerns of end users that should be addressed through technical means. We then illustrate one specific significant problem facing end users, namely that of certificate errors, which can be exploited to facilitate pervasive surveillance. We suggest that users should not be required to determine whether a certificate error is valid, but instead to block access to websites that generate such errors. We believe this approach would be more effective in protecting end users in an environment of persistent network threats.

C.48. Developer-Resistant Cryptography - Kelsey Cairns, Graham Steel

48.pdf [76] - "Properly implemented strong crypto systems are one of the few things that you can rely on" - Edward Snowden. So why is mass surveillance so successful? One (big) problem is endpoint security. Another is that strong crypto systems are sufficiently difficult to implement that often either mistakes are made resulting in catastrophic loss of security, or cryptography is not used at all. What can we do to make cryptography easier to use and more resistant

to developer errors?

C.49. Improving the reliability of key ownership assertions - Kai Engert

49.pdf [77] - A majority of today's secure Internet connections rely on Certificate Authorities not being abused for issueing false certificates (key ownership assertions), which might get abused for interception purposes, despite the risk of detection. I suggest to enhance Internet protocols with protective mechanisms to detect false key ownership assertions. Ideas: (1) Using a network of proxy services, for example as implemented by the The Onion Router (Tor), consistency checking chould be performed by individual clients, in order to detect assertions that are likely false, prior to allowing a connection (see Detector.io). (2) Extend the idea that notary services provide a second opinion about the correctness of key ownership assertions, by requiring CAs to run such services (kuix.de/mecai). (3) Implement protocol extensions, where client software reports previously seen key ownership assertions to the operators of services, allowing the discovery of false ownership assertions.

C.50. Mike O'Neill's Position Paper - Mike O'Neill

50.pdf [78]

C.51. Detecting MITM Attacks on Ephemeral Diffie-Hellman without Relying on a PKI in Real-Time Communications - Alan Johnston

51.pdf [79] - With the recent revelations about pervasive surveillance on the Internet, there is renewed interest in techniques that protect against passive eavesdropping without relying on a Public Key Infrastructure (PKI). An ephemeral Diffie-Hellman (DH) key agreement can provide such protection, but (without authentication) the exchange is vulnerable to a Man in the Middle (MitM) attack. An example of a protocol that has MitM protection for a DH key agreement is ZRTP, RFC 6189, "ZRTP: Media Path Key Agreement for Unicast Secure RTP." ZRTP provides pervasive surveillance resistant security for Voice over IP (VoIP), video communication, and other real-time communication services. This paper describes the techniques used by ZRTP to detect MitM attacks, and explores whether these techniques could be used to develop a general MitM detection protocol to be used by other non-real-time communication protocols. An example of how ZRTP can provide MitM detection for another protocol, DTLS-SRTP, Datagram Transport Layer Security - Secure Real-time Transport Protocol, is given.

C.52. Trust & Usability on the Web, a Social/Legal perspective - Rigo Wenning, Bert Bos

52.pdf [80] - (1) The browsers' UIs for security are very technical and seem to avoid saying anything useful, maybe so that the browsers and CAs cannot be held responsible. (2) A user wanting to configure security has difficulty finding the UI and then often discovers that settings are hard-coded or unclear. (3) The security model is based on trusting a few commercial entities and mistrusting the user, who ends up without control over his software if one of those entities is compromised or doesn't share his goals. Conclusion: We need better UIs, which in turn requires a PKI that has the metadata and social aspects that help users understand and explore the keys and the organizations behind them.

C.53. Hardening Operations and Management Against Passive Eavesdropping - Bernard Aboba

53.pdf [81] - Today within service providers protocols used for operations and management frequently send data in the clear, enabling the data to be collected by passive eavesdroppers. Examples of operations and management protocols include Authentication, Authorization and Accounting (AAA), syslog and Simple Networking Monitoring Protocol (SNMP). Since the publication of "Operational Security Current Practices in Internet Service Provider Environments" [RFC4778], the IETF has developed specifications that enable per-packet confidentiality to be applied to operations and management protocols. By developing updated operational guidance recommending deployment of per-packet confidentiality based on recent IETF Request for Comments (RFCs) and work-in-progress, the IETF can assist in bringing customer and regulatory pressure to bear in improving operational practices.

C.54. A few theses regarding privacy and security - Andreas Kuckartz

54.pdf [82]

C.55. Meet the new threat model, same as the old threat model - Eric Rescorla

55.pdf [83] - 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. This includes being able to generate packets that appear to be from a trusted machine. Thus, even if the end-system with which you wish to communicate is itself secure, the Internet environment provides no assurance that packets which claim to be from that system in fact are. - [RFC3552]

C.56. It's Time for Application-Centric Security - Yuan Gu, Harold Johnson

56.pdf [84] - An 'application' is an organized data/executable-code compound performing a specific function or service. We hold that applications should be protected intrinsically (by obfuscated, tamper-resistant code and data), as well as extrinsically (by encrypted communication, hardened hardware platforms, authenticated access). (1) Cloud-based applications are vulnerable to their hosting services or neighbors. (2) Peripheral-based applications (on phones, pads, PDAs, or more generally in the Internet of Things) are vulnerable because hardware security is inconsistent and very expensive to repair. (3) Browser-based applications are vulnerable because they run on potentially hostile or malware-infected browsers or platforms which we don't control. Application obfuscations such as homomorphic transforms on data and computation (motto: avoid data or computation in plain form) and increased interdependency (motto: aggressive fragility under tampering) can effectively address these vulnerabilities.

C.57. Sabatini Monatesti position paper - Sabatine Monatesti

57.pdf [85]

C.58. Trust problems in pervasive monitoring - Melinda Shore, Karen O'Donoghue

58.pdf [86]

C.59. Beyond "Just TLS Everywhere": From Client-encrypted Messaging to Defending the Social Graph - Harry Halpin, George Danezis

59.pdf [87]

C.60. Network Security as a Public Good - Wendy Seltzer

60.pdf [88] - Network security depends on cooperation of multiple actors in the Internet ecosystem. Standards consortia should support

and help coordinate activity to protect the commons.

C.61. Statement of Interest on behalf of the W3C TAG - Dan Appelquist

61.pdf [89]

C.62. Improving Security on the Internet - Hannes Tschofenig

62.pdf [90] - Securing the Internet has been an on-going activity since the early days of the IETF and a variety of technical specifications have been standardized. Someone reading through IETF RFCs might easily get the impression that the Internet should be very secure. This is, however, not the entire story as the never-ending series of breaches and recently the Snowden revelations have illustrated. While on paper everything looks pretty good many problems can be found in implementations and with deployments.

In this position paper the author makes the argument that improving the collaboration between different communities in the Internet software development life-cycle will be crucial for improving security on the Internet.

C.63. Protecting customer data from government snooping - Orit Levin

63.pdf [91]

C.64. Privacy Aware Internet Development Initiative 2014 - Achim Klabunde

64.pdf [92] - Protecting privacy on the Internet requires more than using encryption. Protocols, implementations and applications must minimise the amount of personal data that is distributed and collected. Work is required to develop and disseminate privacy aware design and impmementation techniques to the actual developers. The paper is a call for interest for an initiative aiming to address this need, supported by privacy and technology experts.

C.65. The Internet is Broken: Idealistic Ideas for Building a NEWGNU Network - Christian Grothoff, Bartlomiej Polot, Carlo von Loesch

65.pdf [93] - This paper describes issues for security and privacy at all layers of the Internet stack and proposes radical changes to the architecture to build a network that offers strong security and privacy by default.

C.66. Opportunistic Keying as a Countermeasure to Pervasive Monitoring - Stephen Kent

66.pdf [94] - This document was prepared as part of the IETF response to concerns about "pervasive monitoring" as articulated in [draft-farrell-perpass-attack]. It begins by exploring terminology that has been used in IETF standards (and in academic publications) to describe encryption and key management techniques, with a focus on authentication vs. anonymity. Based on this analysis, it propose a new term, "opportunistic keying" (OK) to describe a goal for IETF security protocols, one possible countermeasure to pervasive monitoring. It reviews key management mechanisms used in IETF security protocol standards, with respect to these properties, to identify what changes might be needed to offer OK with minimal changes. The document ends by examining possible impediments to and potential adverse effects associated with deployment and use of techniques that would increase the use of encryption, even when keys are distributed in an unauthenticated manner.

Appendix D. Workshop chairs & program committee

The workshop chairs were three: Stephen Farrell [95] (TCD) and Rigo Wenning [96] (W3C) from the STREWS project, and Hannes Tschofenig [97] (ARM) from the STREWS Interest Group.

A program committee (PC) was charged with evaluating the submitted papers. It was made up of the members of the STREWS project, the members of the STREWS Interest Group, plus invited experts: Bernard Aboba (Microsoft), Dan Appelquist (Telefonica & W3C TAG), Richard Barnes (Mozilla), Bert Bos (W3C), Lieven Desmet (KU Leuven), Karen O'Donoghue (ISOC), Russ Housley (Vigil Security), Martin Johns (SAP), Ben Laurie (Google), Eliot Lear (Cisco), Kenny Paterson (Royal Holloway), Eric Rescorla (RTFM), Wendy Seltzer (W3C), Dave Thaler (Microsoft) and Sean Turner (IECA).

Appendix E. Participants

The participants to the workshop were:

- *Bernard Aboba* (Microsoft Corporation)
- *Thijs Alkemade* (Adium)
- *Daniel Appelquist* (Telefonica Digital)

- *Jari Arkko* (Ericsson)
- *Alia Atlas* (Juniper Networks)
- *Emmanuel Baccelli* (INRIA)
- *Mary Barnes*
- *Richard Barnes* (Mozilla)
- *Steve Bellovin* (Columbia University)
- *Andrea Bittau* (Stanford University)
- *Marc Blanchet* (Viagenie)
- *Carsten Bormann* (Uni Bremen TZI)
- *Bert Bos* (W3C)
- *Ian Brown* (Oxford University)
- *Stewart Bryant* (Cisco Systems)
- *Randy Bush* (IIJ / Dragon Research Labs)
- *Kelsey Cairns* (Washington State University)
- *Stuart Cheshire* (Apple)
- *Vincent Cheval* (University of Birmingham)
- *Benoit Claise* (Cisco)
- *Alissa Cooper* (Cisco)
- *Dave Crocker* (Brandenburg InternetWorking)
- *Leslie Daigle* (Internet Society)
- *George Danezis* (University College London)
- *Spencer Dawkins* (Huawei)
- *Mark Donnelly* (Painless Security)
- *Nick Doty* (W3C)

- *Dan Druta* (AT&T)
- *Peter Eckersley* (Electronic Frontier Foundation)
- *Lars Eggert* (NetApp)
- *Kai Engert* (Red Hat)
- *Monika Ermert*
- *Stephen Farrell* (Trinity College Dublin)
- *Barbara Fraser* (Cisco)
- *Virginie Galindo* (gemalto)
- *Stefanie Gerdes* (Uni Bremen TZI)
- *Daniel Kahn Gillmor* (ACLU)
- *Wendy M. Grossman*
- *Christian Grothoff* (The GNUnet Project)
- *Oliver Hahm* (INRIA)
- *Joseph Lorenzo Hall* (Center for Democracy & Technology)
- *Phillip Hallam-Baker*
- *Harry Halpin* (W3C/MIT and IRI)
- *Ted Hardie* (Google)
- *Joe Hildebrand* (Cisco Systems)
- *Russ Housley* (Vigil Security, LLC)
- *Cullen Jennings* (CISCO)
- *Leif Johansson* (SUNET)
- *Harold Johnson* (Irdeto)
- *Alan Johnston* (Avaya)
- *L. Aaron Kaplan* (CERT.at)

- *Steve Kent* (BBN Technologies)
- *Achim Klabunde* (European Data Protection Supervisor)
- *Hans Kuhn* (NOC)
- *Christian de Larrinaga*
- *Ben Laurie* (Google)
- *Eliot Lear* (Cisco Ssytems)
- *Barry Leiba* (Huawei Technologies)
- *Sebastian Lekies* (SAP AG)
- *Orit Levin* (Microsoft Corporation)
- *carlo von lynX* (#youbroketheinternet)
- *Xavier Marjou* (Orange)
- *Larry Masinter* (Adobe)
- *John Mattsson* (Ericsson)
- *Patrick McManus* (Mozilla)
- *Doug Montgomery* (NIST)
- *Kathleen Moriarty* (EMC)
- *Alec Muffett* (Facebook)
- *Suhas Nandakumar* (Cisco Systems)
- *Linh Nguyen* (ERCIM/W3C)
- *Linus Nordberg* (NORDUnet)
- *Mark Nottingham*
- *Karen O'Donoghue* (Internet Society)
- *Piers O'Hanlon* (Oxford Internet Institute)
- *Kenny Paterson* (Royal Holloway, University of London)

- *Jon Peterson* (Neustar)
- *Joshua Phillips* (University of Birmingham)
- *Alfredo Pironti* (INRIA)
- *Dana Polatin-Reuben* (University of Oxford)
- *Prof. Johan Pouwelse* (Delft University of Technology)
- *Max Pritikin* (Cisco)
- *Eric Rescorla* (Mozilla)
- *Pete Resnick* (Qualcomm Technologies, Inc.)
- *Tom Ristenpart* (University of Wisconsin)
- *Andrei Robachevsky* (Internet Society)
- *David Rogers* (Copper Horse)
- *Scott Rose* (NIST)
- *Christine Runnegar* (Internet Society)
- *Philippe De Ryck* (DistriNet - KU Leuven)
- *Peter Saint-Andre* (&yet)
- *Runa A. Sandvik* (Center for Democracy and Technology)
- *Jakob Schlyter* (キレイ)
- *Dr. Jan Seedorf* (NEC Laboratories Europe)
- *Wendy Seltzer* (W3C)
- *Melinda Shore* (No Mountain Software)
- *Dave Thaler* (Microsoft)
- *Brian Trammell* (ETH Zurich)
- *Hannes Tschofenig* (ARM Limited)
- *Sean Turner* (IECA, Inc.)

- *Matthias Waehlisch* (Freie Universitaet Berlin)
- *Greg Walton* (Oxford University)
- *Rigo Wenning* (W3C)
- *Tara Whalen* (Apple Inc.)
- *Greg Wood* (Internet Society)
- *Jiangshan Yu* (University of Birmingham)
- *Aaron Zauner*
- *Dacheng Zhang* (Huawei)
- *Phil Zimmermann* (Silent Circle LLC)
- *Juan-Carlos Zuniga* (InterDigital)

Authors' Addresses

Stephen Farrell
Trinity College, Dublin

Email: stephen.farrell@cs.tcd.ie

Rigo Wenning
World Wide Web Consortium
2004, route des Lucioles
B.P. 93
Sophia-Antipolis 06902
France

Email: rigo@w3.org

Bert Bos
World Wide Web Consortium
2004, route des Lucioles
B.P. 93
Sophia-Antipolis 06902
France

Email: bert@w3org

Marc Blanchet
Viagenie
246 Aberdeen
Quebec, QC G1R 2E1
Canada

Email: Marc.Blanchet@viagenie.ca
URI: http://viagenie.ca

Hannes Tschofenig
ARM Ltd.
110 Fulbourn Rd
Cambridge CB1 9NJ
Great Britain

Email: Hannes.tschofenig@gmx.net
URI: http://www.tschofenig.priv.at