Confidentiality in the Face of Pervasive Surveillance: A Threat Model
and Problem Statement
draft-iab-privsec-confidentiality-threat-01

Abstract

   Documents published in 2013 revealed several classes of pervasive
   surveillance attack on Internet communications. In this document we
   develop a threat model that describes these pervasive attacks. We
   start by assuming a completely passive attacker with an interest in
   undetected, indiscriminate eavesdropping, then expand the threat
   model with a set of verified attacks that have been published. Based
   on this threat model, we discuss the techniques that can be employed
   in Internet protocol design to increase the protocols robustness to
   pervasive surveillance.

Status of This Memo

This Internet-Draft will expire on June 15, 2015.

Copyright Notice

1. Introduction

Starting in June 2013, documents released to the press by Edward Snowden have revealed several operations undertaken by intelligence agencies to exploit Internet communications for intelligence purposes. These attacks were largely based on protocol vulnerabilities that were already known to exist. The attacks were nonetheless striking in their pervasive nature, both in terms of the amount of Internet communications targeted, and in terms of the diversity of attack techniques employed.

To ensure that the Internet can be trusted by users, it is necessary for the Internet technical community to address the vulnerabilities exploited in these attacks [RFC7258]. The goal of this document is to describe more precisely the threats posed by these pervasive attacks, and based on those threats, lay out the problems that need to be solved in order to secure the Internet in the face of those threats.

The remainder of this document is structured as follows. In Section 3, we describe an idealized passive attacket, one which could completely undetectably compromise communications at Internet scale. In Section 4, we provide a brief summary of some attacks that have been disclosed, and use these to expand the assumed capabilities of our idealized attacker. Section 5 describes a threat model based on these attacks, focusing on classes of attack that have not been a focus of Internet engineering to date.

2. Terminology

This document makes extensive use of standard security and privacy terminology; see [RFC4949] and [RFC6973]. Terms used from [RFC6973] include Eavesdropper, Observer, Initiator, Intermediary, Recipient, Attack (in a privacy context), Correlation, Fingerprint, Traffic Analysis, and Identifiability (and related terms). In addition, we use a few terms that are specific to the attacks discussed here:

Pervasive Attack: An attack on Internet communications that makes use of access at a large number of points in the network, or otherwise provides the attacker with access to a large amount of Internet traffic; see [RFC7258]

Observation: Information collected directly from communications by an eavesdropper or observer. For example, the knowledge that <alice@example.com> sent a message to <bob@example.com> via SMTP taken from the headers of an observed SMTP message would be an observation.

Inference: Information extracted from analysis of information collected directly from communications by an eavesdropper or observer. For example, the knowledge that a given web page was accessed by a given IP address, by comparing the size in octets of measured network flow records to fingerprints derived from known sizes of linked resources on the web servers involved, would be an inference.

Collaborator: An entity that is a legitimate participant in a communication, but who provides information about that interaction (keys or data) to an attacker.

Unwitting Collaborator: A collaborator that provides information to the attacker not deliberately, but because the attacker has exploited some technology used by the collaborator.

Key Exfiltration: The transmission of keying material for an encrypted communication from a collaborator to an attacker

Content Exfiltration: The transmission of the content of a communication from a collaborator to an attacker

3. An Idealized Pervasive Passive Attacker

To build a thread model, we first assume a pervasive passive attacker, an indiscriminate eavesdropper on an Internet-attached computer network that

o can observe every packet of all communications at any or every hop
   in any network path between an initiator and a recipient; and

o can observe data at rest in intermediate systems between the
   endpoints controlled by the initiator and recipient; but

o takes no other action with respect to these communications (i.e.,
   blocking, modification, injection, etc.).

This attacker is less capable than those which we know to have
compromised the Internet from press reports, elaborated in Section 4,
but represents the threat to communications privacy by a single
eavesdropper interested in remaining undetectable.

The techniques available to our ideal attacker are direct observation
and inference. Direct observation involves taking information
directly from eavesdropped communications - e.g., URLs identifying
content or email addresses identifying individuals from application-
layer headers. Inference, on the other hand, involves analyzing
eavesdropped information to derive new information from it; e.g.,
searching for application or behavioral fingerprints in observed
traffic to derive information about the observed individual from
them, in absence of directly-observed sources of the same
information. The use of encryption to protect confidentiality is
generally enough to prevent direct observation, assuming
uncompromised encryption implementations and key material, but
provides less complete protection against inference, especially
inference based only on unprotected portions of communications (e.g.
IP and TCP headers for TLS [RFC5246]).

3.1. Information subject to direct observation

Protocols which do not encrypt their payload make the entire content
of the communication available to the idealized attacker along their
path. Following the advice in [RFC3365], most such protocols have a
secure variant which encrypts payload for confidentiality, and these
secure variants are seeing ever-wider deployment. A noteworthy
exception is DNS [RFC1035], as DNSSEC [RFC4033] does not have
confidentiality as a requirement. This implies that all DNS queries
and answers generated by the activities of any protocol are available
to the attacker.

Protocols which imply the storage of some data at rest in
intermediaries (e.g. SMTP [RFC5321]) leave this data subject to
observation by an attacker that has compromised these intermediaries,
unless the data is encrypted end-to-end by the application layer
protocol, or the implementation uses an encrypted store for this
data.

3.2. Information useful for inference

   Inference is information extracted from later analysis of an observed
   or eavesdropped communication, and/or correlation of observed or
   eavesdropped information with information available from other
   sources. Indeed, most useful inference performed by the attacker
   falls under the rubric of correlation. The simplest example of this
   is the observation of DNS queries and answers from and to a source
   and correlating those with IP addresses with which that source
   communicates. This can give access to information otherwise not
   available from encrypted application payloads (e.g., the Host:
   HTTP/1.1 request header when HTTP is used with TLS).

   Protocols which encrypt their payload using an application- or
   transport-layer encryption scheme (e.g. TLS) still expose all the
   information in their network and transport layer headers to the
   attacker, including source and destination addresses and ports.
   IPsec ESP[RFC4303] further encrypts the transport-layer headers, but
   still leaves IP address information unencrypted; in tunnel mode,
   these addresses correspond to the tunnel endpoints. Features of the
   cryptographic protocols themselves, e.g. the TLS session identifier,
   may leak information that can be used for correlation and inference.
   While this information is much less semantically rich than the
   application payload, it can still be useful for the inferring an
   individual's activities.

   Inference can also leverage information obtained from sources other
   than direct traffic observation. Geolocation databases, for example,
   have been developed map IP addresses to a location, in order to
   provide location-aware services such as targeted advertising. This
   location information is often of sufficient resolution that it can be
   used to draw further inferences toward identifying or profiling an
   individual.

   Social media provide another source of more or less publicly
   accessible information. This information can be extremely
   semantically rich, including information about an individual's
   location, associations with other individuals and groups, and
   activities. Further, this information is generally contributed and
   curated voluntarily by the individuals themselves: it represents
   information which the individuals are not necessarily interested in
   protecting for privacy reasons. However, correlation of this social
   networking data with information available from direct observation of
   network traffic allows the creation of a much richer picture of an
   individual's activities than either alone. We note with some alarm
   that there is little that can be done at protocol design time to
   limit such correlation by the attacker, and that the existence of

   such data sources in many cases greatly complicates the problem of
   protecting privacy by hardening protocols alone.

3.3. An illustration of an ideal passive attack

   To illustrate how capable the attacker is even given its limitations,
   we explore the non-anonymity of even encrypted IP traffic by
   examining in detail some inference techniques for associating a set
   of addresses with an individual, in order to illustrate the
   difficulty of defending communications against our idealized
   attacker. Here, the basic problem is that information radiated even
   from protocols which have no obvious connection with personal data
   can be correlated with other information which can paint a very rich
   behavioral picture, that only takes one unprotected link in the chain
   to associate with an identity.

3.3.1. Analysis of IP headers

   Internet traffic can be monitored by tapping Internet links, or by
   installing monitoring tools in Internet routers. Of course, a single
   link or a single router only provides access to a fraction of the
   global Internet traffic. However, monitoring a number of high
   capacity links or a set of routers placed at strategic locations
   provides access to a good sampling of Internet traffic.

   Tools like IPFIX [RFC7011] allow administrators to acquire statistics
   about sequences of packets with some common properties that pass
   through a network device. The most common set of properties used in
   flow measurement is the "five-tuple"of source and destination
   addresses, protocol type, and source and destination ports. These
   statistics are commonly used for network engineering, but could
   certainly be used for other purposes.

   Let's assume for a moment that IP addresses can be correlated to
   specific services or specific users. Analysis of the sequences of
   packets will quickly reveal which users use what services, and also
   which users engage in peer-to-peer connections with other users.
   Analysis of traffic variations over time can be used to detect
   increased activity by particular users, or in the case of peer-to-
   peer connections increased activity within groups of users.

3.3.2. Correlation of IP addresses to user identities

   The correlation of IP addresses with specific users can be done in
   various ways. For example, tools like reverse DNS lookup can be used
   to retrieve the DNS names of servers. Since the addresses of servers
   tend to be quite stable and since servers are relatively less
   numerous than users, an attacker could easily maintain its own copy

of the DNS for well-known or popular servers, to accelerate such
lookups.

On the other hand, the reverse lookup of IP addresses of users is
generally less informative. For example, a lookup of the address
currently used by one author's home network returns a name of the
form "c-192-000-002-033.hsd1.wa.comcast.net". This particular type
of reverse DNS lookup generally reveals only coarse-grained location
or provider information.

In many jurisdictions, Internet Service Providers (ISPs) are required
to provide identification on a case by case basis of the "owner" of a
specific IP address for law enforcement purposes. This is a
reasonably expedient process for targeted investigations, but
pervasive surveillance requires something more efficient. This
provides an incentive for the attacker to secure the cooperation of
the ISP in order to automate this correlation.

3.3.3. Monitoring messaging clients for IP address correlation

Even if the ISP does not cooperate, user identity can often be
obtained via inference. POP3 [RFC1939] and IMAP [RFC3501] are used
to retrieve mail from mail servers, while a variant of SMTP is used
to submit messages through mail servers. IMAP connections originate
from the client, and typically start with an authentication exchange
in which the client proves its identity by answering a password
challenge. The same holds for the SIP protocol [RFC3261] and many
instant messaging services operating over the Internet using
proprietary protocols.

The username is directly observable if any of these protocols operate
in cleartext; the username can then be directly associated with the
source address.

3.3.4. Retrieving IP addresses from mail headers

SMTP [RFC5321] requires that each successive SMTP relay adds a
"Received" header to the mail headers. The purpose of these headers
is to enable audit of mail transmission, and perhaps to distinguish
between regular mail and spam. Here is an extract from the headers
of a message recently received from the "perpass" mailing list:

"Received: from 192-000-002-044.zone13.example.org (HELO
?192.168.1.100?) (xxx.xxx.xxx.xxx) by lvps192-000-002-219.example.net
with ESMTPSA (DHE-RSA-AES256-SHA encrypted, authenticated); 27 Oct
2013 21:47:14 +0100 Message-ID: <526D7BD2.7070908@example.org> Date:
Sun, 27 Oct 2013 20:47:14 +0000 From: Some One <some.one@example.org>
"

This is the first "Received" header attached to the message by the
first SMTP relay; for privacy reasons, the field values have been
anonymized. We learn here that the message was submitted by "Some
One" on October 27, from a host behind a NAT (192.168.1.100)
[RFC1918] that used the IP address 192.0.2.44. The information
remained in the message, and is accessible by all recipients of the
"perpass" mailing list, or indeed by any attacker that sees at least
one copy of the message.

An attacker that can observe sufficient email traffic can regularly
update the mapping between public IP addresses and individual email
identities. Even if the SMTP traffic was encrypted on submission and
relaying, the attacker can still receive a copy of public mailing
lists like "perpass".

3.3.5. Tracking address usage with web cookies

Many web sites only encrypt a small fraction of their transactions.
A popular pattern is to use HTTPS for the login information, and then
use a "cookie" to associate following clear-text transactions with
the user's identity. Cookies are also used by various advertisement
services to quickly identify the users and serve them with
"personalized" advertisements. Such cookies are particularly useful
if the advertisement services want to keep tracking the user across
multiple sessions that may use different IP addresses.

As cookies are sent in clear text, an attacker can build a database
that associates cookies to IP addresses for non-HTTPS traffic. If
the IP address is already identified, the cookie can be linked to the
user identify. After that, if the same cookie appears on a new IP
address, the new IP address can be immediately associated with the
pre-determined identity.

3.3.6. Graph-based approaches to address correlation

An attacker can track traffic from an IP address not yet associated
with an individual to various public services (e.g. websites, mail
servers, game servers), and exploit patterns in the observed traffic
to correlate this address with other addresses that show similar
patterns. For example, any two addresses that show connections to
the same IMAP or webmail services, the same set of favorite websites,
and game servers at similar times of day may be associated with the
same individual. Correlated addresses can then be tied to an
individual through one of the techniques above, walking the "network
graph" to expand the set of attributable traffic.

3.3.7. Tracking of MAC Addresses

Moving back down the stack, technologies like Ethernet or Wi-Fi use
MAC Addresses to identify link-level destinations. MAC Addresses
assigned according to IEEE-802 standards are unique to the device.
If the link is publicly accessible, an attacker can track it. For
example, the attacker can track the wireless traffic at public Wi-Fi
networks. Simple devices can monitor the traffic, and reveal which
MAC Addresses are present. If the network does not use some form of
Wi-Fi encryption, or if the attacker can access the decrypted
traffic, the analysis will also provide the correlation between MAC
Addresses and IP addresses. Additional monitoring using techniques
exposed in the previous sections will reveal the correlation between
MAC Addresses, IP Addresses, and user identity. The attacker can
conceivably build a database linking MAC Addresses and device or user
identities, and use it to track the movement of devices and of their
owners.

4. Reported Instances of Large-Scale Attacks

The situation in reality is more bleak than that suggested by an
analysis of our idealized attacker. Through revelations of sensitive
documents in several media outlets, the Internet community has been
made aware of several intelligence activities conducted by US and UK
national intelligence agencies, particularly the US National Security
Agency (NSA) and the UK Government Communications Headquarters
(GCHQ). These documents have revealed methods that these agencies
use to attack Internet applications and obtain sensitive user
information.

First, they have confirmed that these agencies have capabilities in
line with those of our idealized attacker, thorugh the large-scale
passive collection of Internet traffic [pass1][pass2][pass3][pass4].
For example: - The NSA XKEYSCORE system accesses data from multiple
access points and searches for "selectors" such as email addresses,
at the scale of tens of terabytes of data per day. - The GCHQ
Tempora system appears to have access to around 1,500 major cables
passing through the UK. - The NSA MUSCULAR program tapped cables
between data centers belonging to major service providers. - Several
programs appear to perform wide-scale collection of cookies in web
traffic and location data from location-aware portable devices such
as smartphones.

However, the capabilities described go beyond those available to our
idealized attacker, including:

o Decryption of TLS-protected Internet sessions [dec1][dec2][dec3].
   For example, the NSA BULLRUN project appears to have had a budget

of around $250M per year to undermine encryption through multiple
approaches.

o Insertion of NSA devices as a man-in-the-middle of Internet
   transactions [TOR1][TOR2]. For example, the NSA QUANTUM system
   appears to use several different techniques to hijack HTTP
   connections, ranging from DNS response injection to HTTP 302
   redirects.

o Direct acquisition of bulk data and metadata from service
   providers [dir1][dir2][dir3]. For example, the NSA PRISM program
   provides the agency with access to many types of user data (e.g.,
   email, chat, VoIP).

o Use of implants (covert modifications or malware) to undermine
   security and anonymity features [dec2][TOR1][TOR2]. For example:

   * NSA appears to use the QUANTUM man-in-the-middle system to
      direct users to a FOXACID server, which delivers an implant to
      compromise the browser of a user of the Tor anonymous
      communications network.

   * The BULLRUN program mentioned above includes the addition of
      covert modifications to software as one means to undermine
      encryption.

   * There is also some suspicion that NSA modifications to the
      DUAL_EC_DRBG random number generator were made to ensure that
      keys generated using that generator could be predicted by NSA.
      These suspicions have been reinforced by reports that RSA
      Security was paid roughly $10M to make DUAL_EC_DRBG the default
      in their products.

We use the term "pervasive attack" [RFC7258] to collectively describe
these operations. The term "pervasive" is used because the attacks
are designed to indiscriminately gather as much data as possible and
to apply selective analysis on targets after the fact. This means
that all, or nearly all, Internet communications are targets for
these attacks. To achieve this scale, the attacks are physically
pervasive; they affect a large number of Internet communications.
They are pervasive in content, consuming and exploiting any
information revealed by the protocol. And they are pervasive in
technology, exploiting many different vulnerabilities in many
different protocols.

It's important to note that although the attacks mentioned above were
executed by NSA and GCHQ, there are many other organizations that can
mount pervasive surveillance attacks. Because of the resources

required to achieve pervasive scale, these attacks are most commonly
undertaken by nation-state actors. For example, the Chinese Internet
filtering system known as the "Great Firewall of China" uses several
techniques that are similar to the QUANTUM program, and which have a
high degree of pervasiveness with regard to the Internet in China.

5. Threat Model

Given these disclosures, we must consider a broader threat model.

Pervasive surveillance aims to collect information across a large
number of Internet communications, analyzing the collected
communications to identify information of interest within individual
communications, or inferring information from correlated
communications. his analysis sometimes benefits from decryption of
encrypted communications and deanonymization of anonymized
communications. As a result, these attackers desire both access to
the bulk of Internet traffic and to the keying material required to
decrypt any traffic that has been encrypted. Even if keys are not
available, note that the presence of a communication and the fact
that it is encrypted may both be inputs to an analysis, even if the
attacker cannot decrypt the communication.

The attacks listed above highlight new avenues both for access to
traffic and for access to relevant encryption keys. They further
indicate that the scale of surveillance is sufficient to provide a
general capability to cross-correlate communications, a threat not
previously thought to be relevant at the scale of the Internet.

5.1. Attacker Capabilities

Attack Class	Capability
Passive observation	Directly capture data in transit
Passive inference	Infer from reduced/encrypted data
Active	Manipulate / inject data in transit
Static key exfiltration	Obtain key material once / rarely
Dynamic key exfiltration	Obtain per-session key material
Content exfiltration	Access data at rest

Security analyses of Internet protocols commonly consider two classes
of attacker: Passive attackers, who can simply listen in on
communications as they transit the network, and active attackers, who
can modify or delete packets in addition to simply collecting them.

In the context of pervasive passive surveillance, these attacks take
on an even greater significance. In the past, these attackers were
often assumed to operate near the edge of the network, where attacks
can be simpler. For example, in some LANs, it is simple for any node
to engage in passive listening to other nodes' traffic or inject
packets to accomplish active attacks. However, as we now know, both
passive and active attacks are undertaken by pervasive attackers
closer to the core of the network, greatly expanding the scope and
capability of the attacker.

Eavesdropping and observation at a larger scale make passive
inference attacks easier to carry out: a passive attacker with access
to a large portion of the Internet can analyze collected traffic to
create a much more detailed view of individual behavior than an
attacker that collects at a single point. Even the usual claim that
encryption defeats passive attackers is weakened, since a pervasive
passive attacker can infer relationships from correlations over large
numbers of sessions, e.g., pairing encrypted sessions with
unencrypted sessions from the same host, or performing traffic
fingerprinting between known and unknown encrypted sessions. Reports
on the NSA XKEYSCORE system would indicate it is an example of such
an attacker.

A pervasive active attacker likewise has capabilities beyond those of
a localized active attacker. Active attacks are often limited by
network topology, for example by a requirement that the attacker be
able to see a targeted session as well as inject packets into it. A
pervasive active attacker with access at multiple points within the
core of the Internet is able to overcome these topological
limitations and perform attacks over a much broader scope. Being
positioned in the core of the network rather than the edge can also
enable a pervasive active attacker to reroute targeted traffic,
amplifying the ability to perform both eavesdropping and traffic
injection. Pervasive active attackers can also benefit from
pervasive passive collection to identify vulnerable hosts.

While not directly related to pervasiveness, attackers that are in a
position to mount a pervasive active attack are also often in a
position to subvert authentication, a traditional protection against
such attacks. Authentication in the Internet is often achieved via
trusted third party authorities such as the Certificate Authorities
(CAs) that provide web sites with authentication credentials. An
attacker with sufficient resources may also be able to induce an

authority to grant credentials for an identity of the attacker's
choosing. If the parties to a communication will trust multiple
authorities to certify a specific identity, this attack may be
mounted by suborning any one of the authorities (the proverbial
"weakest link"). Subversion of authorities in this way can allow an
active attack to succeed in spite of an authentication check.

Beyond these three classes (observation, inference, and active),
reports on the BULLRUN effort to defeat encryption and the PRISM
effort to obtain data from service providers suggest three more
classes of attack:

o Static key exfiltration

o Dynamic key exfiltration

o Content exfiltration

These attacks all rely on a collaborator providing the attacker with
some information, either keys or data. These attacks have not
traditionally been considered in security analyses of protocols,
since they happen outside of the protocol.

The term "key exfiltration" refers to the transfer of keying material
for an encrypted communication from the collaborator to the attacker.
By "static", we mean that the transfer of keys happens once, or
rarely, typically of a long-lived key. For example, this case would
cover a web site operator that provides the private key corresponding
to its HTTPS certificate to an intelligence agency.

"Dynamic" key exfiltration, by contrast, refers to attacks in which
the collaborator delivers keying material to the attacker frequently,
e.g., on a per-session basis. This does not necessarily imply
frequent communications with the attacker; the transfer of keying
material may be virtual. For example, if an endpoint were modified
in such a way that the attacker could predict the state of its
psuedorandom number generator, then the attacker would be able to
derive per-session keys even without per-session communications.

Finally, content exfiltration is the attack in which the collaborator
simply provides the attacker with the desired data or metadata.
Unlike the key exfiltration cases, this attack does not require the
attacker to capture the desired data as it flows through the network.
The risk is to data at rest as opposed to data in transit. This
increases the scope of data that the attacker can obtain, since the
attacker can access historical data - the attacker does not have to
be listening at the time the communication happens.

Exfiltration attacks can be accomplished via attacks against one of
the parties to a communication, i.e., by the attacker stealing the
keys or content rather than the party providing them willingly. In
these cases, the party may not be aware that they are collaborating,
at least at a human level. Rather, the subverted technical assets
are "collaborating" with the attacker (by providing keys/content)
without their owner's knowledge or consent.

Any party that has access to encryption keys or unencrypted data can
be a collaborator. While collaborators are typically the endpoints
of a communication (with encryption securing the links),
intermediaries in an unencrypted communication can also facilitate
content exfiltration attacks as collaborators by providing the
attacker access to those communications. For example, documents
describing the NSA PRISM program claim that NSA is able to access
user data directly from servers, where it is stored unencrypted. In
these cases, the operator of the server would be a collaborator, if
an unwitting one. By contrast, in the NSA MUSCULAR program, a set of
collaborators enabled attackers to access the cables connecting data
centers used by service providers such as Google and Yahoo. Because
communications among these data centers were not encrypted, the
collaboration by an intermediate entity allowed NSA to collect
unencrypted user data.

5.2. Attacker Costs

Attack Class	Cost / Risk to Attacker
Passive observation	Passive data access
Passive inference	Passive data access + processing
Active	Active data access + processing
Static key exfiltration	One-time interaction
Dynamic key exfiltration	Ongoing interaction / code change
Content exfiltration	Ongoing, bulk interaction

Each of the attack types discussed in the previous section entails
certain costs and risks. These costs differ by attack, and can be
helpful in guiding response to pervasive attack.

Depending on the attack, the attacker may be exposed to several types
of risk, ranging from simply losing access to arrest or prosecution.

In order for any of these negative consequences to occur, however,
the attacker must first be discovered and identified. So the primary
risk we focus on here is the risk of discovery and attribution.

A passive attack is the simplest to mount in some ways. The base
requirement is that the attacker obtain physical access to a
communications medium and extract communications from it. For
example, the attacker might tap a fiber-optic cable, acquire a mirror
port on a switch, or listen to a wireless signal. The need for these
taps to have physical access or proximity to a link exposes the
attacker to the risk that the taps will be discovered. For example,
a fiber tap or mirror port might be discovered by network operators
noticing increased attenuation in the fiber or a change in switch
configuration. Of course, passive attacks may be accomplished with
the cooperation of the network operator, in which case there is a
risk that the attacker's interactions with the network operator will
be exposed.

In many ways, the costs and risks for an active attack are similar to
those for a passive attack, with a few additions. An active attacker
requires more robust network access than a passive attacker, since
for example they will often need to transmit data as well as
receiving it. In the wireless example above, the attacker would need
to act as an transmitter as well as receiver, greatly increasing the
probability the attacker will be discovered (e.g., using direction-
finding technology). Active attacks are also much more observable at
higher layers of the network. For example, an active attacker that
attempts to use a mis-issued certificate could be detected via
Certificate Transparency [RFC6962].

In terms of raw implementation complexity, passive attacks require
only enough processing to extract information from the network and
store it. Active attacks, by contrast, often depend on winning race
conditions to inject pakets into active connections. So active
attacks in the core of the network require processing hardware to
that can operate at line speed (roughly 100Gbps to 1Tbps in the core)
to identify opportunities for attack and insert attack traffic in a
high-volume traffic.
Key exfiltration attacks rely on passive attack for access to
encrypted data, with the collaborator providing keys to decrypt the
data. So the attacker undertakes the cost and risk of a passive
attack, as well as additional risk of discovery via the interactions
that the attacker has with the collaborator.

In this sense, static exfiltration has a lower risk profile than
dynamic. In the static case, the attacker need only interact with
the collaborator a small number of times, possibly only once, say to
exchange a private key. In the dynamic case, the attacker must have

continuing interactions with the collaborator. As noted above these
interactions may real, such as in-person meetings, or virtual, such
as software modifications that render keys available to the attacker.
Both of these types of interactions introduce a risk that they will
be discovered, e.g., by employees of the collaborator organization
noticing suspicious meetings or suspicious code changes.

Content exfiltration has a similar risk profile to dynamic key
exfiltration. In a content exfiltration attack, the attacker saves
the cost and risk of conducting a passive attack. The risk of
discovery through interactions with the collaborator, however, is
still present, and may be higher. The content of a communication is
obviously larger than the key used to encrypt it, often by several
orders of magnitude. So in the content exfiltration case, the
interactions between the collaborator and the attacker need to be
much higher-bandwidth than in the key exfiltration cases, with a
corresponding increase in the risk that this high-bandwidth channel
will be discovered.

It should also be noted that in these latter three exfiltration
cases, the collaborator also undertakes a risk that his collaboration
with the attacker will be discovered. Thus the attacker may have to
incur additional cost in order to convince the collaborator to
participate in the attack. Likewise, the scope of these attacks is
limited to case where the attacker can convince a collaborator to
participate. If the attacker is a national government, for example,
it may be able to compel participation within its borders, but have a
much more difficult time recruiting foreign collaborators.

As noted above, the collaborator in an exfiltration attack can be
unwitting; the attacker can steal keys or data to enable the attack.
In some ways, the risks of this approach are similar to the case of
an active collaborator. In the static case, the attacker needs to
steal information from the collaborator once; in the dynamic case,
the attacker needs to continued presence inside the collaborators
systems. The main difference is that the risk in this case is of
automated discovery (e.g., by intrusion detection systems) rather
than discovery by humans.

[6]. Security Considerations

This document describes a threat model for pervasive surveillance
attacks. Mitigations are to be given in a future document.

7. IANA Considerations

This document has no actions for IANA.

8. Acknowledgements

Thanks to Dave Thaler for the list of attacks and taxonomy; to
Security Area Directors Stephen Farrell, Sean Turner, and Kathleen
Moriarty for starting and managing the IETF's discussion on pervasive
attack; and to Stephan Neuhaus, Mark Townsley, Chris Inacio, and
Evangelos Halepilidis, and to the membership of the IAB Privacy and
Security Program for their input.

9. References

9.1. Normative References

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J.,
           Morris, J., Hansen, M., and R. Smith, "Privacy
           Considerations for Internet Protocols", RFC 6973, July
           2013.

9.2. Informative References

[pass1]   The Guardian, "How the NSA is still harvesting your online
           data", 2013,
           <http://www.theguardian.com/world/2013/jun/27/
           nsa-online-metadata-collection>.

[pass2]   The Guardian, "NSA's Prism surveillance program: how it
           works and what it can do", 2013,
           <http://www.theguardian.com/world/2013/jun/08/
           nsa-prism-server-collection-facebook-google>.

[pass3]   The Guardian, "XKeyscore: NSA tool collects 'nearly
           everything a user does on the internet'", 2013,
           <http://www.theguardian.com/world/2013/jul/31/
           nsa-top-secret-program-online-data>.

[pass4]   The Guardian, "How does GCHQ's internet surveillance
           work?", n.d., <http://www.theguardian.com/uk/2013/jun/21/
           how-does-gchq-internet-surveillance-work>.

[dec1]   The New York Times, "N.S.A. Able to Foil Basic Safeguards
           of Privacy on Web", 2013,
           <http://www.nytimes.com/2013/09/06/us/
           nsa-foils-much-internet-encryption.html>.

   [dec2]   The Guardian, "Project Bullrun - classification guide to
               the NSA's decryption program", 2013,
               <http://www.theguardian.com/world/interactive/2013/sep/05/
               nsa-project-bullrun-classification-guide>.

   [dec3]   The Guardian, "Revealed: how US and UK spy agencies defeat
               internet privacy and security", 2013,
               <http://www.theguardian.com/world/2013/sep/05/
               nsa-gchq-encryption-codes-security>.

   [TOR]   The Tor Project, "Tor", 2013,
               <https://www.torproject.org/>.

   [TOR1]   Schneier, B., "How the NSA Attacks Tor/Firefox Users With
               QUANTUM and FOXACID", 2013,
               <https://www.schneier.com/blog/archives/2013/10/
               how_the_nsa_att.html>.

   [TOR2]   The Guardian, "'Tor Stinks' presentation - read the full
               document", 2013,
               <http://www.theguardian.com/world/interactive/2013/oct/04/
               tor-stinks-nsa-presentation-document>.

   [dir1]   The Guardian, "NSA collecting phone records of millions of
               Verizon customers daily", 2013,
               <http://www.theguardian.com/world/2013/jun/06/
               nsa-phone-records-verizon-court-order>.

   [dir2]   The Guardian, "NSA Prism program taps in to user data of
               Apple, Google and others", 2013,
               <http://www.theguardian.com/world/2013/jun/06/
               us-tech-giants-nsa-data>.

   [dir3]   The Guardian, "Sigint - how the NSA collaborates with
               technology companies", 2013,
               <http://www.theguardian.com/world/interactive/2013/sep/05/
               sigint-nsa-collaborates-technology-companies>.

   [secure]   Schneier, B., "NSA surveillance: A guide to staying
               secure", 2013,
               <http://www.theguardian.com/world/2013/sep/05/
               nsa-how-to-remain-secure-surveillance>.

   [snowden]   Technology Review, "NSA Leak Leaves Crypto-Math Intact but
               Highlights Known Workarounds", 2013,
               <http://www.technologyreview.com/news/519171/nsa-leak-
               leaves-crypto-math-intact-but-highlights-known-
               workarounds/>.

25

[key-recovery]
          Golle, P., "The Design and Implementation of Protocol-
          Based Hidden Key Recovery", 2003,
          <http://crypto.stanford.edu/~pgolle/papers/escrow.pdf>.

[RFC1035] Mockapetris, P., "Domain names - implementation and
           specification", STD 13, RFC 1035, November 1987.

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and
           E. Lear, "Address Allocation for Private Internets", BCP
           5, RFC 1918, February 1996.

[RFC1939] Myers, J. and M. Rose, "Post Office Protocol - Version 3",
           STD 53, RFC 1939, May 1996.

[RFC2015] Elkins, M., "MIME Security with Pretty Good Privacy
           (PGP)", RFC 2015, October 1996.

[RFC2821] Klensin, J., "Simple Mail Transfer Protocol", RFC 2821,
           April 2001.

[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.

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION
           4rev1", RFC 3501, March 2003.

[RFC3851] Ramsdell, B., "Secure/Multipurpose Internet Mail
           Extensions (S/MIME) Version 3.1 Message Specification",
           RFC 3851, July 2004.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S.
           Rose, "DNS Security Introduction and Requirements", RFC
           4033, March 2005.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the
           Internet Protocol", RFC 4301, December 2005.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC
           4303, December 2005.

   [RFC4306]   Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC
               4306, December 2005.

   [RFC4949]   Shirey, R., "Internet Security Glossary, Version 2", RFC
               4949, August 2007.

   [RFC5246]   Dierks, T. and E. Rescorla, "The Transport Layer Security
               (TLS) Protocol Version 1.2", RFC 5246, August 2008.

   [RFC5321]   Klensin, J., "Simple Mail Transfer Protocol", RFC 5321,
               October 2008.

   [RFC5655]   Trammell, B., Boschi, E., Mark, L., Zseby, T., and A.
               Wagner, "Specification of the IP Flow Information Export
               (IPFIX) File Format", RFC 5655, October 2009.

   [RFC5750]   Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet
               Mail Extensions (S/MIME) Version 3.2 Certificate
               Handling", RFC 5750, January 2010.

   [RFC6120]   Saint-Andre, P., "Extensible Messaging and Presence
               Protocol (XMPP): Core", RFC 6120, March 2011.

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

   [RFC6698]   Hoffman, P. and J. Schlyter, "The DNS-Based Authentication
               of Named Entities (DANE) Transport Layer Security (TLS)
               Protocol: TLSA", RFC 6698, August 2012.

   [RFC7011]   Claise, B., Trammell, B., and P. Aitken, "Specification of
               the IP Flow Information Export (IPFIX) Protocol for the
               Exchange of Flow Information", STD 77, RFC 7011, September
               2013.

   [RFC7258]   Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an
               Attack", BCP 188, RFC 7258, May 2014.

Authors' Addresses

   Richard Barnes

   Email: rlb@ipv.sx

   Bruce Schneier

   Email: schneier@schneier.com

Cullen Jennings

Email: fluffy@cisco.com

Ted Hardie

Email: ted.ietf@gmail.com

Brian Trammell

Email: ietf@trammell.ch

Christian Huitema

Email: huitema@huitema.net

Daniel Borkmann

Email: dborkman@redhat.com