Introduction to the Science and Politics of Modern Internet Encryption

Brian C. Tracy
CS1805 — Brown University

December 2019

1 Introduction

Modern encryption techniques allow people to send data that is both secure and private over communication channels that are fundamentally insecure and non-private. This capability is the foundation of critical Internet services such as online banking, e-commerce, and instant messaging. As privacy becomes a mainstream consumer focus, more people are encrypting their Internet traffic[1]. However, the increasing adoption of strong encryption has made governments worldwide nervous as it poses a threat to the state's ability to monitor its citizens. This paper aims to serve as an introduction to understanding the relationships between governments, the laws they create and enforce, private corporations, and most importantly, citizens when discussing Internet encryption.

The first section will briefly describe how the modern Internet works and how communications flow through it. Next, the issue known as "Going Dark" is described though the lens of several historical case studies and the interests and motivations of the many parties who have a stake in the issue will be summarized. After this, the current state of legislation in the United States will be discussed. Finally, my personal musings on the subject will be put to the page.

2 Internet Fundamentals

The physical, theoretical, and computational limits of what is possible to achieve with computers serves as a starting point for understanding the legislature that surrounds their usage. A basic knowledge of the technical details behind Internet encryption is useful in understanding where certain laws fall on the spectrum from being trivially enforceable to physically impossible to enforce.

[1] Adrienne Porter Felt et al, *Measuring HTTPS Adoption on the Web*, 9

2.1 HTTP

At the heart of virtually all[2] everyday exchanges of information on the World Wide Web lies the **H**yper**t**ext **T**ransfer **P**rotocol. Proposed in 1999[3], HTTP provides a standardized communication format for the transportation of web pages and other web related content.

The most common use case for HTTP is mediating a conversation between a “client” and a “server”. In most scenarios, a web browser (Safari, Chrome, Firefox, . . .) acts as a client on the behalf of a user, and the server is a computer owned by whoever the user is attempting to communicate with (Amazon, Google, Wikipedia, . . .). In this client-server model, the client sends “requests” and the server sends “responses”. Below is an example of a client requesting the home page from `www.example.com`

```
GET /homepage.html HTTP/1.1
host: www.example.com
```

To which the server owned by `www.example.com` might respond

```
HTTP/1.1 200 OK
Date: Wed, 13 Nov 2019 01:21:56 GMT
Content-Length: 37

<html><body>Hello, this is example.com’s homepage</body></html>
```

In the minimal example above, the important parts of the request are the resource being requested (the homepage identified by `/homepage.html`) and the host name (the value of the `host` field).

The response starts with a status code (`200 OK`) which indicates whether or not the requested resource could be found. Next comes a series of fields that contain information about the resource being requested. Finally, the web page itself is included at the end of the response.

The majority of interactions on the web are simply a chain of request/response pairs that are orchestrated by user interaction such as clicking links or submitting forms.

Receiving information from a server is only one type of interaction that can be mediated by HTTP. A common use case for modern web applications is submitting information to the server so that it can do something on your behalf. The simplest example of this is providing your username and password to your bank so that it may log you in and display your balance.

When submitting information to a server, a client can use the HTTP `POST` action and include the information to be submitted at the end of the request.

```
POST /get-balance.html HTTP/1.1
host: my-bank.com

username = david_smith
password = sandcastles
```

[2] Dozens of other protocols exist, but the usages this paper focus on such as online banking and messaging are conducted over HTTP

[3] R. Fielding et al, *Hypertext Transfer Protocol – HTTP/1.1*, 1

After the bank's server receives this request, it would perform some verification, then respond with the user's balance.

```
HTTP/1.1 200 OK
Date: Wed, 13 Nov 2019 01:45:56 GMT
Content-Length: 72

<html><body>Welcome Dave, you have 400USD in your account.</body></html>
```

Notice how both the user's password, and his account balance are visible within the request/response pair. This presents a problem for banking applications and is the driving motivation behind the encryption of Internet traffic. Ideally, only Dave and his bank should be able to see the sensitive information being transmitting during their sessions. The next section will briefly illustrate why this is difficult using just the HTTP protocol.

2.2 Binary Format of HTTP

HTTP is designed to be a human readable protocol, meaning that the contents of an HTTP request/response, as well as all the other information such as version numbers and headers, are represented as readable characters. However, to be transmitted over a network, the text of the request/response needs to be encoded into a stream of numbers that computers can understand. Below is a hexdump[4] of an example HTTP request. The column on the right is what the bytes[5] on the left decode to if they are to be interpreted as text. Notice that not every byte translates to an english character. This is because there are 256 different values that one byte can represent, but there are less than half as many alphanumeric and punctuation characters.

```
00 08 e3 ff fc 08 6c 96 cf da d2 29 08 00 45 00    ......l....)..E.
00 7f 1a 35 40 00 40 06 dd 72 0a 26 02 d1 5d b8    ...5@.@..r.&..].
d8 22 e0 a4 00 50 4c b0 df df dc e8 d7 0b 80 18    ."...PL.........
10 08 05 31 00 00 01 01 08 0a 5d 9c fb 93 9a b6    ...1......].....
c8 26 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31    .&GET / HTTP/1.1
0d 0a 48 6f 73 74 3a 20 65 78 61 6d 70 6c 65 2e    ..Host: example.
63 6f 6d 0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a    com..User-Agent:
20 63 75 72 6c 2f 37 2e 35 34 2e 30 0d 0a 41 63     curl/7.54.0..Ac
63 65 70 74 3a 20 2a 2f 2a 0d 0a 0d 0a             cept: */*....
```

The data above, when considered as a single unit, is known as a "packet". The portions of the packet that are not the encoded HTTP request represent data that is necessary for routing and transmitting the packet. This can be considered "metadata" when discussing the difference between the intented communication (the HTTP request), and the entirety of the packet.

[4] A readout of a sequence of bytes, encoded as pairs of base 16 "hexadecimal" numbers

[5] 1 byte = 8 bits. A bit is either a 1 or 0. A byte being composed of 8 bits means that there are 2^8 possible values that a byte can represent. For the sake of convenience, bytes are often shown as a pair of hexadecimal digits (0 through 9, then A through F), therefore the smallest numerical value of a byte would be 00, and the largest FF

2.3 Routing

The process of finding a path between the client and the server along which to send the information of any given network transaction is called "routing". Encoded within the packet's metadata is the packet's destination IP address (in the above example, it is the four byte sequence near the beginning `[5d b8 d8 22]`). This IP address identifies the server to which the request is being sent. In addition, the packet also contains a source IP, which identifies the sender of the packet. These IP addresses are not necessarily uniquely translatable to a human identity, but in some cases they can be used to narrow down a sender to a specific house or device.

As a packet is sent across the Internet, at each stopping point it passes through a piece of hardware known as a switch that has complete access to the entirety of the packet. This means that whoever is responsible for routing an individual packet knows where it came from, where it is being sent, and the full contents of the packet. Looking back at the bank scenario, this is problematic because when traversing the Internet, packets can take several stops at locations that the sender of the packet has no control over.

This aspect of the routing process is why Internet communication is fundamentally insecure and non private. Any switch along the path can read or modify the packets it is responsible for routing. The owner of these switches is most often an Internet Service Provider (ISP), but it could also be the government in a "municipal broadband" situation. In either case, the users of the network have to entrust their traffic to a third party, which is simply unacceptable.

2.4 HTTPS

To avoid leaving user privacy in the hands of ISPs, the contents of modern Internet communications are encrypted through a process known as Public Key Cryptography. The cryptographic systems discussed below operate on sequences of bytes, which can be fairly unwieldy to talk about. For this reason, certain byte sequences that have special purposes are given more practical and meaningful names like "key", "message", and "certificate".

At the center of a Public Key Cryptography system lies the "key pair", which is simply a pair of cryptographic keys, one designated as the "private key", the other designated as the "public key". The private key is kept a secret, and the public key is distributed to anyone who asks for it. These keys are carefully generated such that the following property holds[6]:

The private and public keys are "inverses". This means that messages encrypted with one can only be decrypted by the other.

The proof of the above claim is very involved and will not be included. However, once accepted to be true[7] the power of the result is immense as the following observations can be made.

1. If I give you my public key, and you encrypt a message with it, then only I can decrypt your message.

[6] Whitfield Diffie and Susan Landau, *Privacy on the Line*, 39

[7] This is not much too much to ask of the reader, the smartest minds in the field have proven the result.

2. If I encrypt a message with my private key, anyone who decrypts that encrypted message with my public key can be certain that I was the author, for only my public key can decrypt messages that were created with my private key.

Using the first observation, the sending and receiving of encrypted messages can be explained. If I wish to send a secret message to my neighbor, all I have to do is encrypt my message with his public key. Only his private key can decrypt this message, and because he is (read: should be) the only one with access to his private key, only he can read my message.

Secure communication is accomplished via employing public and private keys in tandem. However, the issue of trust is not addressed with this scheme. By its definition, the public key is available for everyone to use. This means that anyone can send my neighbor a message encrypted with his public key and pretend to be me. To sort this out, a third party is necessary to mediate the exchange of keys to ensure that the people "on the other end of the line" are who they claim to be.

2.5 Certificate Authorities

The Certificate Authority (CA) is a trusted third party that offers assurances about the identities that claim to own a certain public key. In the consumer world, this is essential because there needs to be a way to prove that the owner of `www.bankofamerica.com` (and its corresponding public key) is actually the real Bank of America corporation.

Being labeled as a "trusted third party" carries a huge amount of responsibility because so far, it is the only part of the encryption scheme that needs to be accepted as true by everyone involved. In practice, companies like DigiCert and other CAs make their money off of earning this trust.

To confirm the identity of a public key's holder, a CA will provide proof in the form of a "digital certificate". The previous section's second observation makes this to be possible. If a trusted CA makes the claim of "www.bankofamerica.com is legitimate", and encrypts this message with its private key, then anyone who decrypts the statement with the CA's public key can be sure that the statement came from the CA and nobody else. This proof of claim by a CA is a digital certificate, and it is issued to server owners. So, if Bank of America has a digital certificate from a trusted CA, they can safely announce (and be trusted when doing so) that their public key actually belongs to them.

Strong Public Key Cryptography combined with robust Certificate Authorities makes HTTPS capable of providing a trustworthy and private means of communication over the Internet.

2.6 Adoption Rates

Historically, there have been several factors preventing the widespread adoption of HTTPS. Foremost is the issue of certificate management. Computer system administrators who wish to serve their content over HTTPS must ensure that all their servers have access to valid certificates. Valid certificates need to be issued by trusted CAs (this costs money), and they need to be kept up to date as they expire after a certain amount of time.

In addition, the necessity of HTTPS is not seen by all server owners. For example, in applications with no possible security concerns for their users do not strictly require HTTPS. For sites where users do not have to log in or provide any sensitive information, there is no need for the traffic to be encrypted. Also, the cryptographic operations required by HTTPS impose a slight computational overhead on every connection that a server has to handle.

Let's Encrypt is a non-profit organization that eliminates many of the hassles that come with adopting HTTPS. As per their mission statement, "Let's Encrypt is a free, automated, and open certificate authority"[8]. By providing certificates for free, and more importantly, managing the expiration of the issued certificates, Let's Encrypt makes upgrading to HTTPS trivial for many server owners. Their work, along with an increasingly privacy aware population, has lead to an increase in adoption of HTTPS.

Google, the developer of the Chrome browser, has a large interest in increasing the amount of HTTPS traffic on the Internet. In accordance with their long term efforts to promote HTTPS, they also release reports on the status of adoption. Among the top 100 websites not owned by Google, HTTPS usage increased from 39% to 54% between 2016 and 2017[9].

From a strictly security and privacy point of view, the increase in adoption of strong encryption via HTTPS is a great accomplishment. Consumers are now safer than ever when performing sensitive operations over the Internet. However, anyone attempting to carry out large scale data collection is not happy about the dwindling number of plain text HTTP connections.

3 Going Dark

> *Thank you for the opportunity to testify today about the growing challenges to public safety and national security that have eroded our ability to obtain electronic information and evidence pursuant to a court order or warrant. We in law enforcement often refer to this problem as "Going Dark." - James Comey*[10]

With HTTPS rapidly becoming the standard for Internet communication, government wiretaps are being made useless. When the ISPs themselves are not capable of knowing what information is flowing over their networks, they cannot comply with lawful warrants to produce the information in an unencrypted form.

When implemented correctly, encryption prevents anyone but the two parties in communication from knowing what is being communicated. This is an unacceptable fact for many governments, so they insist that encryption be done incorrectly.

Unless otherwise stated, "the government" in question is that of the United States. This simplification serves to narrow down the scope of the research, but causes little to be lost in terms of conclusions drawn. As the global hub of the Internet, US law has widespread international effects on Internet users.

[8] Let's Encrypt Homepage

[9] Adrienne Porter Felt et al, *Measuring HTTPS Adoption on the Web*, 9

[10] James B. Comey, *Encryption Tightrope: Balancing Americans' Security and Privacy*

3.1 Government Objections to Encryption

One of the many roles of government is to keep its citizens safe. Speaking to this mission, FBI Director James Comey said in 2016 that "Investigating and prosecuting [cyber criminals, child pornographers, fraudsters, . . .] is a core responsibility and priority of the Department of Justice"[11]. A major tool in the arsenal of law enforcement for carrying out its mission is the warrant. While the warrant has had a long and technical legal history in the United States, not all of the issues surrounding the issuing and enforcement of warrants have been solved. Specifically, modern encryption presents a new challenge to the theoretical possibility of complying with warrants.

Recall that when set up correctly, encrypted traffic over the Internet can only be decrypted by the sender and receiver of the data. So, despite being responsible for carrying the traffic between parties, the ISP is unable to crack open the packets and see what is being said. This makes it impossible (if best practices are being followed by the ISP, CA, . . .) for an ISP to hand over the data it transports. According to current Deputy Attorney General Rod Rosenstein, this creates a "law-free zones that permit criminals and terrorists to operate without detection by police and without accountability by judges and juries"[12].

So, despite hundreds of years of precedent supporting the government's right to serve and enforce warrants, this ability is completely eliminated by modern encryption. In its current form, modern encryption is "warrant proof" in ways that physical security measures are not. For example, any arguments surrounding the right of a government to break a lock on a backyard shed or a safe deposit box do not transfer to encrypted digital data. We currently do not have the means to break digital locks, regardless of whether it would be legal.

3.2 Possible Solution Space

A main reason why the "Going Dark" problem is ongoing is that obvious solutions are unacceptable and more nuanced approaches hard to find. The government admits to this lack of progress and instead of proposing a fix, the responsibility to comply with regulation is pushed to the recipients of warrants. On the subject of government level solutions, Director Comey stated "We don't have any silver bullet [sic], and the discussions within the Executive Branch are still ongoing"[13].

It needs to be mentioned that the "obvious solution" to "Going Dark" is to make all encrypted communications illegal. It would become trivial for the ISPs being served warrants to hand over communications and the hundreds of years of legal precedent surrounding warrants could be employed to ensure that people's private information is not being wrongfully seized by the government. However, just because the government at times "plays by the rules" in terms of acquiring communication data, bad actors have no incentive to do so. Without all of the benefits of modern encryption, Internet communication goes back to being insecure and useless for many critical applications. The government admits to the economic infeasibility of this approach and it is not a realistic goal. However, banning encryption

[11] James B. Comey, *Encryption Tightrope: Balancing Americans' Security and Privacy*

[12] Rod J. Rosenstein, *Deputy Attorney General Rod J. Rosenstein Delivers Remarks on Encryption at the United States Naval Academy*, 4

[13] James B. Comey, *Encryption Tightrope: Balancing Americans' Security and Privacy*

outright serves as one end of the spectrum of solutions.

The current situation falls on the opposite end of the solution spectrum. Laws banning encryption have not yet arrived, and the usage of encryption is increasing. By starting a discussion around "Going Dark", the government hopes to come across a solution somewhere between these two realities – the present, and the reality where there is no encryption.

4 Government Proposed Solutions

The most infamous of government solutions was the "Clipper Chip" and the associated "Key Escrow System". Announced in 1993[14], the Clipper Chip was a computer chip designed to enable secure voice communication. The caveat was that the private keys hardwired into each chip were not actually *private* keys. Each Clipper Chip's private key would also be shared with the government with the promise that it was only to be used to decrypt the communications after a valid warrant was issued.

The Clipper Chip was doomed to failure from its very inception. Private keys are private for a reason: cryptographic algorithms have no concept of "intent". From a computational perspective, there is no difference between the government lawfully using their copy of a private key to decrypt communications and a hacker using a stolen copy of that private key. Having a centralized repository (the key escrow) of private keys meant that a single compromised database renders every communications device using a Clipper Chip insecure.

The Clipper Chip was abandoned (read: rejected) by the tech community in 1996 after security flaws were discovered in the implementation of the Key Escrow system that allowed parties that were not the government to decrypt communications passing through the chips. This failure was the ultimate proof of a fact long known by the security community: If one person has a backdoor into a secure system, it is only a matter of time before everybody has that backdoor.

5 Legal History

The current state of encrypted communication ensures that ISPs cannot know the unencrypted meaning of the data they transmit. For example, below is an excerpt of a packet very similar to that of the one included in a previous section. The previous packet was generated by issuing a request to `http://example.com`, while the excerpt below comes from a secure HTTPS request issued to `https://example.com`

```
00 08 e3 ff fc 08 6c 96 cf da d2 29 08 00 45 00   ......l....)..E.
c2 38 30 95 39 e9 3a b2 3f 2a 90 52 44 00 00 44   .80.9.:.?*.RD..D
00 9e 00 6b 00 67 00 39 00 33 00 16 00 9d 00 9c   ...k.g.9.3......
0b 65 78 61 6d 70 6c 65 2e 63 6f 6d 00 0a 00 08   .example.com....
00 06 00 17 00 18 00 19 00 0b 00 02 01 00 00 0d   ................
00 12 00 10 04 01 02 01 05 01 06 01 04 03 02 03   ................
```

[14] CryptoMuseum.org, *Clipper Chip, Cryptographic Key Escrow*

Notice how, due to the content of the packet being encrypted, the actual data being sent is no longer human readable. This is exactly what encrypted communication should look like, and attempting to decipher the contents of the packet is a hopeless task.

If the government were to serve an ISP with a warrant and demand to know what was being communicated between my computer and example.com, it would be perfectly reasonable for the ISP to provide the above information. In fact, the ISP could preemptively allow the government to inspect *all* traffic flowing through their network, and the security of the system would not be compromised.

For this reason, the legal argument surrounding Internet encryption is not one of simply procuring the data – it is already visible to all, yet entirely useless. The more interesting issue is whether or not the government can *compel* ISPs to provide the *unencrypted* data in response to a lawful warrant.

5.1 CALEA

The **C**ommunications **A**ssistance for **L**aw **E**nforcement **A**ct lays out the requirements that telecommunication providers must follow to allow the government to conduct digital surveillance[15]. Routes to CALEA compliance are not explicitly stated by the government, and for good reason. Electronic communications take many forms, and having to specify the exact details of allowing them to be intercepted would be impossible. Instead, the FCC offers suggestions to service providers on how they could approach becoming CALEA compliant. Three of the options presented are to develop an in house solution, pass the responsibility off to the manufacturer of the hardware that is being used, or purchase a solution from a third party. At the end of the day, under CALEA, *someone* has to be responsible for providing the impossible – that is, decrypted communication data.

5.2 Technical Assistance

The mechanism by which the government has grounds to compel telecommunication companies to do the impossible[16] is found in a piece of legislature commonly known as the "Wiretap Act".

> *An order authorizing the interception of a wire [...] shall furnish the applicant forthwith all information, facilities, and* ***technical assistance necessary*** *to accomplish the interception [...]* [17]

So there exist legal grounds on which the government can force ISPs to alter their systems to become CALEA compliant. This is only currently possible through drastic measures such as Clipper Chips, or private Key Escrows. In addition, the government could require the cooperation of a Certificate Authority to achieve a similar outcome. With the help of a CA, the government could give themselves digital certificates that trick Internet users into trusting government owned public keys standing in for real public keys. As an example, the

[15] FCC, *Communications Assistance for Law Enforcement Act*

[16] Currently impossible, that is, before any changes are made to the overall system

[17] 18 U.S.C. §2518(4)(e).

government could compel DigiCert to sign a fake public key for Bank of America, and then substitute this fake key in for the real one. This kind of attack is known as a “Man in the Middle” (MITM).

In order for ISPs to be CALEA compliant with respect to encrypted traffic, they would have to perform a MITM attack on every connection passing through their network, and then provide the decrypted communications upon request.

6 Reality

Despite the legal precedent in place, the scenario above simply cannot be allowed to happen. Similarly to the Clipper Chip outcome, if one party has the power to decrypt all communication, it is only a matter of time before that power falls into the wrong hands.

6.1 Industry

The private sector has compelling reasons to resist CALEA compliance with respect to encrypted data. Messaging applications such as Signal, WhatsApp, and Facebook Messenger exist solely to provide secure communication capabilities to their customers. A major selling point of these applications is that the communications they facilitate are end to end encrypted. Having the power to comply with government requests for data would go against the core philosophy of the application.

6.2 Current Events

The might of the ISP and tech industry has not yet been tested by that of the government. There exists a state of détente between the two factions that has been pushed towards a final verdict, but never forced to the Supreme Court. Clashes between the government and the private sector have been resolved out of court, such as the interaction between the FBI and Apple, but no ruling has been made.

6.3 Opinions

My personal opinion on the subject is that any encroachment from the government on the ability to encrypt Internet traffic would be a terrible mistake. The current encryption scheme is quite binary – it is either secure for everyone, or completely broken. From a utilitarian standpoint, there does not exist a criminal threat, be it cyber attacks, terrorism, or child pornography, that outweighs the benefits that encryption bring. The binary nature of the modern encryption system forces us to choose between increased protection from the criminal threat and the elimination of secure e-commerce. This is not a realistic choice to me.

References

[1] Adrienne Porter Felt et al, *Measuring HTTPS Adoption on the Web*, 2017, https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/46197.pdf

[2] R. Fielding et al, *Hypertext Transfer Protocol – HTTP/1.1*, The Internet Society, 1999, https://tools.ietf.org/html/rfc2616

[3] Whitfield Diffie and Susan Landau, *Privacy on the Line*, The MIT Press, 1998

[4] Let's Encrypt Homepage, 2019, https://letsencrypt.org/

[5] James B. Comey, *Encryption Tightrope: Balancing Americans' Security and Privacy*, Statement to Congress, 2016, https://www.fbi.gov/news/testimony/encryption-tightrope-balancing-americans-security-and-privacy

[6] Rod J. Rosenstein, *Deputy Attorney General Rod J. Rosenstein Delivers Remarks on Encryption at the United States Naval Academy*, 2017, Class Canvas Page

[7] CryptoMuseum.org, *Clipper Chip, Cryptographic Key Escrow*, 2018, https://www.cryptomuseum.com/crypto/usa/clipper.htm

[8] FCC, *Communications Assistance for Law Enforcement Act*, 2019, https://www.fcc.gov/public-safety-and-homeland-security/policy-and-licensing-division/general/communications-assistance