TLS 1.3

Nick Sullivan @grittygrease

Filippo Valsorda @FiloSottile
1994 — SSLv2
1995 — SSLv3
1999 — TLS 1.0
2006 — TLS 1.1
2008 — TLS 1.2

...
PHYSICAL CONNECTIONS

Phone

Data Center

PC

Router

SPs

IX

Cloudflare
**TLS 1.2 ECDHE**

- **Client Hello**
  - Supported cipher suites

- **Server Hello**
  - Chosen cipher suite
  - Key share
  - Certificate & signature

- **Key share**

- **Finished**

- **HTTP GET**

- **HTTP Answer**

- **Finished**
## TLS 1.2 ECDHE

<table>
<thead>
<tr>
<th>Time</th>
<th>Source</th>
<th>Destin</th>
<th>Protocol</th>
<th>Length</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>1051.721834</td>
<td>10.21</td>
<td>104.4</td>
<td>TLSv1.2</td>
<td>251</td>
<td>Client Hello</td>
</tr>
<tr>
<td>1051.747907</td>
<td>104.4</td>
<td>10.21</td>
<td>TLSv1.2</td>
<td>1404</td>
<td>Server Hello</td>
</tr>
<tr>
<td>1051.747993</td>
<td>104.4</td>
<td>10.21</td>
<td>TLSv1.2</td>
<td>811</td>
<td>Certificate Status, Server Key Exchange, Server Hello Done</td>
</tr>
<tr>
<td>1051.749915</td>
<td>10.21</td>
<td>104.4</td>
<td>TLSv1.2</td>
<td>170</td>
<td>Client Key Exchange, Change Cipher Spec, Hello Request, Hello Request</td>
</tr>
<tr>
<td>1051.774692</td>
<td>104.4</td>
<td>10.21</td>
<td>TLSv1.2</td>
<td>302</td>
<td>New Session Ticket, Change Cipher Spec, Encrypted Handshake Message</td>
</tr>
<tr>
<td>1051.774846</td>
<td>104.4</td>
<td>10.21</td>
<td>TLSv1.2</td>
<td>113</td>
<td>Application Data</td>
</tr>
<tr>
<td>1052.104322</td>
<td>10.21</td>
<td>104.4</td>
<td>TLSv1.2</td>
<td>97</td>
<td>Application Data</td>
</tr>
<tr>
<td>1052.104447</td>
<td>10.21</td>
<td>104.4</td>
<td>TLSv1.2</td>
<td>100</td>
<td>Application Data</td>
</tr>
<tr>
<td>1052.104574</td>
<td>10.21</td>
<td>104.4</td>
<td>TLSv1.2</td>
<td>86</td>
<td>Application Data</td>
</tr>
<tr>
<td>1052.104984</td>
<td>10.21</td>
<td>104.4</td>
<td>TLSv1.2</td>
<td>82</td>
<td>Application Data</td>
</tr>
<tr>
<td>1052.105458</td>
<td>10.21</td>
<td>104.4</td>
<td>TLSv1.2</td>
<td>540</td>
<td>Application Data</td>
</tr>
<tr>
<td>1052.128825</td>
<td>104.4</td>
<td>10.21</td>
<td>TLSv1.2</td>
<td>82</td>
<td>Application Data</td>
</tr>
<tr>
<td>1052.132494</td>
<td>104.4</td>
<td>10.21</td>
<td>TLSv1.2</td>
<td>449</td>
<td>Application Data</td>
</tr>
<tr>
<td>1052.132561</td>
<td>104.4</td>
<td>10.21</td>
<td>TLSv1.2</td>
<td>82</td>
<td>Application Data</td>
</tr>
</tbody>
</table>
TLS 1.0, 1.1 and 1.2 are not that different

TLS 1.3 is a BIG jump
RTT--;
TLS 1.3

**Client**

**Client Hello**
- Supported AEAD / groups / signatures
- Key share

**Finished**

**HTTP GET**

**Server**

**Server Hello**
- Chosen AEAD
- Key share

**Certificate & signature**

**Finished**

**HTTP Answer**

HTTP GET
## TLS 1.3

<table>
<thead>
<tr>
<th>Time</th>
<th>Source</th>
<th>Destin</th>
<th>Protocol</th>
<th>Length</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.819792</td>
<td>10.1...</td>
<td>104...</td>
<td>TLSv1.3</td>
<td>561</td>
<td>Client Hello</td>
</tr>
<tr>
<td>3.848558</td>
<td>104...</td>
<td>10....</td>
<td>TLSv1.3</td>
<td>1404</td>
<td>Server Hello, Application Data</td>
</tr>
<tr>
<td>3.849548</td>
<td>104...</td>
<td>10....</td>
<td>TLSv1.3</td>
<td>580</td>
<td>Application Data, Application Data</td>
</tr>
<tr>
<td>3.849608</td>
<td>104...</td>
<td>10....</td>
<td>TLSv1.3</td>
<td>93</td>
<td>Application Data</td>
</tr>
<tr>
<td>3.851004</td>
<td>10.1...</td>
<td>104...</td>
<td>TLSv1.3</td>
<td>102</td>
<td>Application Data</td>
</tr>
<tr>
<td>3.875635</td>
<td>104...</td>
<td>10....</td>
<td>TLSv1.3</td>
<td>221</td>
<td>Application Data</td>
</tr>
</tbody>
</table>
Resumption

“Hey, I know you!”
Client Hello
Supported cipher suites

Key share
Finished

HTTP GET

Server Hello
Session ID

New Session Ticket
Finished

HTTP Answer
TLS 1.2 Resumption

Client Hello
Session ID / Ticket

Server Hello
Finished

Finished
HTTP GET

HTTP Answer
TLS 1.3 Resumption

Client Hello
Session Ticket (PSK)

Server Hello
Finished

Finished
HTTP GET

HTTP Answer
Forward Secrecy

Client Hello
Session Ticket (PSK)

Server Hello
Finished

Finished
HTTP GET

Decrypt this with the session ticket key

HTTP Answer
PSK-ECDHE

Client

Client Hello
Session Ticket (PSK)
Key share

Finished
HTTP GET

Server

Server Hello
Key share
Finished

HTTP Answer
0-RTT!
0-RTT

Client

- Client Hello
- Session Ticket (PSK)
- Key share
- HTTP GET

Server

- Server Hello
- Key share
- Finished
- HTTP Answer

Finished
0-RTT!

But...
0-RTT

No PSK-ECDHE
0-RTT w/ ECDHE

Client Hello
Session Ticket (PSK)
Key share
HTTP GET

Server Hello
Key share
Finished
HTTP Answer

Forward secret from here

Client

Server
TLS 1.2 is forward secret:

- Relatively to the certificate: **always** (using ECDHE)
- Relatively to the ticket key: **never**

TLS 1.3 is forward secret:

- Relatively to the certificate: **always**
- Relatively to the ticket key: **except 0-RTT early data** (w/ PSK-ECDHE)
0-RTT

Replays
0-RTT replay

Client Hello
Session Ticket (PSK)
Key share

HTTP GET

Client Hello
Session Ticket (PSK)
Key share

HTTP GET
obfuscated_ticket_age

• The client sends the age in milliseconds of the ticket
• The server checks it matches its view, with some leeway
• Obfuscated with a ticket_age_add value sent as part of the New Session Ticket message

```c
struct {
    opaque identity<1..2^16-1>;
    uint32 obfuscated_ticket_age;
} PskIdentity;
```
0-RTT confirmation

Client Hello
Session Ticket (PSK)
Key share
HTTP POST

Server Hello
Key share
Finished

Finished
HTTP POST

HTTP Answer
max_early_data_size

- The server must either *accept* or *reject* the early data, entirely, without knowing how much there will be.
- If it accepts it and can’t process it, it must *buffer* it.
- Once the Finished comes, all early data is confirmed.
- `max_early_data_size` limits the buffer size.
- Devised with Drew Springall.
It’s the application’s responsibility

Protocols MUST NOT use 0-RTT data without a profile that defines its use.
It’s the API’s responsibility

• Default to 1-RTT
• Allow the server to reject / wait for the Finished
• Let the client to decide what to send in the early data
HTTP and 0-RTT

• Utopia: **GET** is idempotent!

• Reality: nope.

GET /send_money.php?to=filippo&amount=1000
HTTP and 0-RTT

- Utopia: **GET** is idempotent!
- Reality: nope.
HTTP and 0-RTT

Thai Duong, Thiago Valverde, Quan Nguyen
Google Security Team
{thaidn, valverde, quannguyen}@google.com

Never, never, never, never give up.

Winston Churchill
Complexity

Benefit
No Forward Secrecy
To: IETF TLS 1.3 Working Group Members

My name is Andrew Kennedy and I work at BITS, the technology policy division of the Financial Services Roundtable (http://www.fsroundtable.org/bits). My organization represents approximately 100 of the top 150 US-based financial services companies including banks, insurance, consumer finance, and asset management firms.

[...]

Deprecation of the RSA key exchange in TLS 1.3 will cause significant problems for financial institutions, almost all of whom are running TLS internally and have significant, security-critical investments in out-of-band TLS decryption.

[...]
Out-of-band TLS decryption?

Yes, please!
Hi Andrew,

My view concerning your request: no.

Rationale: We're trying to build a more secure internet.

Meta-level comment:

You're a bit late to the party. We're metaphorically speaking at the stage of emptying the ash trays and hunting for the not quite empty beer cans.

More exactly, we are at draft 15 and RSA key transport disappeared from the spec about a dozen drafts ago. I know the banking industry is usually a bit slow off the mark, but this takes the biscuit.

Cheers,

Kenny
Network Working Group
Internet-Draft
Intended status: Informational
Expires: May 1, 2017

Data Center use of Static Diffie-Hellman in TLS 1.3
<draft-green-tls-static-dh-in-tls13-00>
RC4
3DES
AES-CBC

Vaudenay 2002
Boneh/Brumley 2003
BEAST 2011
Lucky13 2013
POODLE 2014
Lucky Microseconds 2015
RSA-PKCS1-1.5

Bleichenbacher 1998(!!)

Jager 2015

DROWN 2016
Compression

CRIME 2012
Renegotiation

Marsh Ray Attack 2009
Renegotiation DoS 2011
Triple Handshake 2014

Replaced with lightweight key update
simplify

& Fortify
Nonce-Disrespecting Adversaries: Practical Forgery Attacks on GCM in TLS

Hanno Böck* Aaron Zauner‡ Sean Devlin$ Juraj Somorovsky† Philipp Jovanovic

May 17, 2016
TLS 1.2 Certificate Authentication

• Cipher negotiation protected by Finished Message (MAC)
• MAC algorithm determined by cipher negotiation

• FREAK, LogJam, CurveSwap: choose weak parameters
TLS 1.2 ECDHE

Client

Client Hello

- Supported cipher suites

Key share

Finished

HTTP GET

Server

Server Hello

- Chosen cipher suite
  - Key share
  - Certificate & signature

NOT SIGNED

Finished

HTTP Answer

HTTP GET
TLS 1.3

Client

**Client Hello**
- Supported AEAD / groups / signatures
- Key share

**Finished**
- HTTP GET

Server

**Server Hello**
- Chosen AEAD
- Key share

**Certificate**
**Signature**
**Finished**

**HTTP Answer**

HTTP GET
Fewer, better choices

- Key Exchange, Cipher, Authentication negotiated separately
- No arbitrary DH groups
- No arbitrary curves
## American Dish

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>Fried Chicken Wings(4)</td>
<td>$5.15</td>
</tr>
<tr>
<td>A2</td>
<td>Buffalo Wings</td>
<td>$5.75</td>
</tr>
<tr>
<td>A3</td>
<td>Fried Chicken Teriyaki(4)</td>
<td>$5.95</td>
</tr>
<tr>
<td>A4</td>
<td>Fried Baby Shrimp(12)</td>
<td>$5.25</td>
</tr>
<tr>
<td>A5</td>
<td>Fried Chicken Scampi</td>
<td>$5.95</td>
</tr>
<tr>
<td>A6</td>
<td>French Fries</td>
<td>$7.15</td>
</tr>
</tbody>
</table>

---

## Appetizers

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Roasted Pork Egg Roll(Each)</td>
<td>$1.20</td>
</tr>
<tr>
<td>2</td>
<td>Shrimp Egg Roll(Each)</td>
<td>$1.35</td>
</tr>
<tr>
<td>5</td>
<td>Vegetable Spring Roll(5)</td>
<td>$1.35</td>
</tr>
<tr>
<td>6</td>
<td>Beef Egg Roll(2)</td>
<td>$2.50</td>
</tr>
<tr>
<td>7</td>
<td>Fried Pork Wonton(10)</td>
<td>$2.75</td>
</tr>
<tr>
<td>8</td>
<td>B-B-Q Spare Ribs(S.95)(13.25)</td>
<td>$4.95</td>
</tr>
<tr>
<td>9</td>
<td>Vegetable Spare Ribs</td>
<td>$5.25</td>
</tr>
<tr>
<td>10</td>
<td>Pu Pu Platter</td>
<td>$10.95</td>
</tr>
</tbody>
</table>

---

## Seafood

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>Roasted Pork Egg Yum(3)</td>
<td>$6.75</td>
</tr>
<tr>
<td>41</td>
<td>Roasted Egg Yum(3)</td>
<td>$6.75</td>
</tr>
<tr>
<td>42</td>
<td>Chicken Egg Yum(3)</td>
<td>$6.75</td>
</tr>
<tr>
<td>43</td>
<td>Beef Egg Yum(3)</td>
<td>$7.75</td>
</tr>
<tr>
<td>44</td>
<td>Vegetable Egg Yum</td>
<td>$7.75</td>
</tr>
<tr>
<td>45</td>
<td>Pu Pu Yum</td>
<td>$10.95</td>
</tr>
</tbody>
</table>

---

## Soup

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>Wonton Soup</td>
<td>$1.75</td>
</tr>
<tr>
<td>16</td>
<td>Egg Drop Soup</td>
<td>$2.00</td>
</tr>
<tr>
<td>17</td>
<td>Chicken Rice Noodle Soup</td>
<td>$1.75</td>
</tr>
<tr>
<td>18</td>
<td>Won Ton Egg Drop Mix</td>
<td>$1.95</td>
</tr>
<tr>
<td>19</td>
<td>Hot &amp; Sour Soup</td>
<td>$2.15</td>
</tr>
<tr>
<td>20</td>
<td>Chicken &amp; Shrimp Soup</td>
<td>$4.75</td>
</tr>
<tr>
<td>21</td>
<td>Vegetable Soup Bean Curd</td>
<td>$1.75</td>
</tr>
<tr>
<td>22</td>
<td>House Special Soup</td>
<td>$5.15</td>
</tr>
</tbody>
</table>

---

## Chow Mein

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>89</td>
<td>SEAFOOD DELIGHT................11.95</td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>CRISPY CHICKEN...................8.95</td>
<td></td>
</tr>
<tr>
<td>91</td>
<td>FAMILY.........................2.40</td>
<td></td>
</tr>
<tr>
<td>92</td>
<td>HONEY &amp; CRISPY...................9.95</td>
<td></td>
</tr>
<tr>
<td>93</td>
<td>CRISPY VEGETABLES................8.95</td>
<td></td>
</tr>
<tr>
<td>94</td>
<td>GENERAL TSO'S CHICKEN...........8.95</td>
<td></td>
</tr>
<tr>
<td>95</td>
<td>SIZZLING SCALLOPS................10.25</td>
<td></td>
</tr>
</tbody>
</table>

---

## Beef

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>57</td>
<td>Beef w. Mixed Vegetables</td>
<td>$8.85</td>
</tr>
<tr>
<td>58</td>
<td>Beef w. Mushrooms</td>
<td>$8.85</td>
</tr>
<tr>
<td>59</td>
<td>Beef w. Onions</td>
<td>$8.85</td>
</tr>
<tr>
<td>60</td>
<td>Beef w. Broccoli</td>
<td>$8.85</td>
</tr>
<tr>
<td>61</td>
<td>Human Beef</td>
<td>$8.85</td>
</tr>
<tr>
<td>62</td>
<td>Human Beef</td>
<td>$8.85</td>
</tr>
<tr>
<td>63</td>
<td>Human Beef</td>
<td>$8.85</td>
</tr>
<tr>
<td>64</td>
<td>Human Beef</td>
<td>$8.85</td>
</tr>
<tr>
<td>65</td>
<td>Human Beef</td>
<td>$8.85</td>
</tr>
<tr>
<td>66</td>
<td>Human Beef</td>
<td>$8.85</td>
</tr>
</tbody>
</table>

---

## Combination Platter

Serve w. Pork Egg Roll & Fried Rice or White Rice

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>67</td>
<td>Chicken w. Cashew Nuts</td>
<td>$9.95</td>
</tr>
<tr>
<td>68</td>
<td>Chicken w. Cashew Nuts</td>
<td>$9.95</td>
</tr>
<tr>
<td>69</td>
<td>Chicken w. Cashew Nuts</td>
<td>$9.95</td>
</tr>
<tr>
<td>70</td>
<td>Chicken w. Black Bean Scum</td>
<td>$9.95</td>
</tr>
<tr>
<td>71</td>
<td>Chicken w. Black Bean Scum</td>
<td>$9.95</td>
</tr>
<tr>
<td>72</td>
<td>Chicken w. Black Bean Scum</td>
<td>$9.95</td>
</tr>
<tr>
<td>73</td>
<td>Human Chicken</td>
<td>$8.95</td>
</tr>
<tr>
<td>74</td>
<td>Human Chicken</td>
<td>$8.95</td>
</tr>
<tr>
<td>75</td>
<td>Human Chicken</td>
<td>$8.95</td>
</tr>
<tr>
<td>76</td>
<td>Human Chicken</td>
<td>$8.95</td>
</tr>
</tbody>
</table>

---

## Side Order

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>77</td>
<td>Fortune Cookies(4)</td>
<td>$0.50</td>
</tr>
<tr>
<td>78</td>
<td>Fortune Cookies(4)</td>
<td>$0.50</td>
</tr>
<tr>
<td>79</td>
<td>Fortune Cookies(4)</td>
<td>$0.50</td>
</tr>
</tbody>
</table>

---

## Dessert

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>Mixed Vegetables.............</td>
<td>$5.95</td>
</tr>
<tr>
<td>81</td>
<td>Mixed Vegetables.............</td>
<td>$5.95</td>
</tr>
<tr>
<td>82</td>
<td>Sauteed Broccoli............</td>
<td>$5.95</td>
</tr>
<tr>
<td>83</td>
<td>Beef w. Cashew Nuts..........</td>
<td>$5.95</td>
</tr>
<tr>
<td>84</td>
<td>Bean Curd w. Garlic Sauce...</td>
<td>$7.15</td>
</tr>
<tr>
<td>85</td>
<td>Bean Curd w. Garlic Sauce...</td>
<td>$7.15</td>
</tr>
<tr>
<td>86</td>
<td>Hot Oil.......................</td>
<td>$0.20</td>
</tr>
<tr>
<td>87</td>
<td>Hot Oil.......................</td>
<td>$0.20</td>
</tr>
</tbody>
</table>

---

## Vegetable

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>96</td>
<td>MONGOLIAN BEEF..............</td>
<td>$9.95</td>
</tr>
<tr>
<td>97</td>
<td>FOUR SEASONS..................</td>
<td>$9.95</td>
</tr>
<tr>
<td>98</td>
<td>DOUBLE DELIGHT W. GARLIC SAUCE</td>
<td>$9.95</td>
</tr>
<tr>
<td>99</td>
<td>SCALLOP &amp; SHRIMP W. GARLIC SAUCE</td>
<td>$10.25</td>
</tr>
<tr>
<td>100</td>
<td>SESAME CHICKEN..............</td>
<td>$9.95</td>
</tr>
<tr>
<td>101</td>
<td>SESAME CHICKEN..............</td>
<td>$9.95</td>
</tr>
<tr>
<td>102</td>
<td>SESAME CHICKEN..............</td>
<td>$9.95</td>
</tr>
<tr>
<td>103</td>
<td>SESAME CHICKEN..............</td>
<td>$9.95</td>
</tr>
<tr>
<td>104</td>
<td>SESAME CHICKEN..............</td>
<td>$9.95</td>
</tr>
<tr>
<td>105</td>
<td>ORANGE CHICKEN..............</td>
<td>$9.95</td>
</tr>
<tr>
<td>106</td>
<td>HONEY CHICKEN..............</td>
<td>$9.95</td>
</tr>
</tbody>
</table>

---

## Fish

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Egg Rolls w. Order of $20 or More only for dinner</td>
<td>FREE</td>
</tr>
<tr>
<td>3</td>
<td>Egg Rolls or Soda(2L) w. Order of $30 or More only for dinner</td>
<td>FREE</td>
</tr>
<tr>
<td>4</td>
<td>Egg Rolls or Soda(2L) w. Order of $40 or More only for dinner</td>
<td>FREE</td>
</tr>
<tr>
<td>5</td>
<td>General's Chicken w. Order of $55 or More only for dinner</td>
<td>FREE</td>
</tr>
</tbody>
</table>
ENTREE
BABY COS SALAD
WITH MARINATED CHICKEN STRIPS, TOASTED BREAD AND CLASSIC CAESAR DRESSING
SHREDDED DUCK SALAD
WITH SHIITAKE MUSHROOMS, SPRING ONION AND BEAN THREAD NOODLES

MAINS
SIRLOIN OF BEEF
WITH ROASTED POTATOES, TOMATO, ONION, ROSEMARY AND GARLIC COMPOTE
GRILLED CHICKEN BREAST
WITH CARAMELISED ONION, MASH POTATOES, TOMATOES, PEAS AND JUS

DESSERT
STICKY DATE PUDDING
WITH BUTTERSCOTCH SAUCE, VANILLA ICECREAM AND SWEET PASTRY
CHOCOLATE MACADAMIA BROWNIE
WITH CHOCOLATE GARNISH AND COCONUT ICE CREAM
Cipher Suites (19 suites)
  Cipher Suite: Unknown (0xdada)
  Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
  Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
  Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)

Extension: signature_algorithms (len=20)
  Type: signature_algorithms (13)
  Length: 20
  Signature Hash Algorithms Length: 18
  Signature Hash Algorithms (9 algorithms)

Extension: elliptic_curves (len=10)
  Type: elliptic_curves (10)
  Length: 10
  Elliptic Curves Length: 8
  Elliptic curves (4 curves)
    Elliptic curve: Unknown (0x9a9a)
    Elliptic curve: ecdh_x25519 (0x001d)
    Elliptic curve: secp256r1 (0x0017)
    Elliptic curve: secp384r1 (0x0018)
Safer Resumption

TLS 1.2 tickets

• **Current** session keys encrypted with session ticket key
• Session ticket key compromise a risk for all connections

TLS 1.3 tickets

• **Next** session keys encrypted with session ticket key
• Session ticket key compromise only risk for resumed connections
TLS 1.2 ECDHE

Client

Client Hello
Supported cipher suites

Key share

Finished

HTTP GET

Unencrypted

Server

Server Hello
Session ID

New Session Ticket

Finished

HTTP Answer
Formal Verification

- Tamarin (Oxford, Royal Holloway)
- ProScript-TLS, miTLS (INRIA)
- nqsb-TLS (Cambridge)
Standards

The IETF way
TLS 1.3 Wish List - it's a short list.
ietf.org/proceedings/87 ...
Timeline

• First Draft: April 17, 2014
  • 3Shake, POODLE, FREAK, LogJam, DROWN, Lucky Microseconds, SLOTH, more...
• Draft 18: October 26, 2016
• Final draft: February, 2017 (we hope)

• TLS 1.2: 79 pages
• TLS 1.3: 81 pages (minus references and appendices)
Transport Layer Security working group of the IETF Discussion Archive - Date Index

[Prev Page] [Next Page] [Thread Index] [IETF Main Archive Directory]

- Dec 22 2016
  - Re: [TLS] Using both External PSK and (EC)DH in TLS 1.3, Eric Rescorla
  - Re: [TLS] Using both External PSK and (EC)DH in TLS 1.3, Russ Houssley
  - Re: [TLS] Using both External PSK and (EC)DH in TLS 1.3, Joseph Salowey
  - Re: [TLS] Using both External PSK and (EC)DH in TLS 1.3, David Benjamin
  - [TLS] Using both External PSK and (EC)DH in TLS 1.3, Russ Houssley
Key Schedule

- Inspired by QUIC crypto
- Semi-static DH key shared out of band
- Tree-based key schedule
PSK -> HKDF-Extract
v
0

+-----> Derive-Secret() = early_traffic_secret

(EC)DHE -> HKDF-Extract
v

+-----> Derive-Secret() = handshake_traffic_secret

0 -> HKDF-Extract
v

+-----> Derive-Secret() = traffic_secret_0

+-----> Derive-Secret() = resumption_master_secret
What's in a name?

Is it TLS 1.3, TLS 2, TLS 2.0, TLS 4, TLS 7, TLS 2017?
Whitney Merrill @wbm312 · Aug 31
What should the next version of TLS (after TLS 1.2) be called? ietf.org/mail-archive/w ... 1,105 votes • Final results

39% TLS 1.3
28% TLS 2.0
13% TLS/2
20% TLS 3

Nick Sullivan @grittygrease · Nov 29
What should the next version of TLS be called? (it's currently called TLS 1.3)
545 votes • Final results

34% TLS 1.3
27% TLS 2
15% TLS 4
24% TLS 7

Ryan Hurst @rmhrsk · Aug 31
What should the new version of TLS be called? #ssl #bikeshedding #standardsbycomitee
322 votes • Final results

21% TLS/42
27% TLS/Bacon
52% Mmmm Bacon
Version Intolerance

- Wire versions
  - SSL 3.0: 3.0
  - TLS 1.0: 3.1
  - TLS 1.1: 3.2
  - TLS 1.2: 3.3
  - TLS 1.3: 3.4 ???

- Servers are intolerant of 3.4
  - >2% of servers fail connection
  - Solution: “3.3” in ClientHello, real versions in extension
  - GREASE by David Benjamin
Version Intolerance

Extension: supported_versions (len=11)
Type: supported_versions (43)
Length: 11
Supported Versions length: 10
Supported Versions: Unknown (0x8a8a)
Supported Versions: TLS 1.3 (draft 18) (0x7f12)
Supported Versions: TLS 1.2 (0x0303)
Supported Versions: TLS 1.1 (0x0302)
Supported Versions: TLS 1.0 (0x0301)
Implementation

Getting our hands dirty
IETF 95 Hackathon - April 2016

- NSS (C): Martin Thomson and Eric Rescorla
- Mint (Go): Richard Barnes and Nick Sullivan

Result:

Firefox was able to load https://tls13.cloudflare.com!
Based on Go crypto/tls

Server only

Audited

crypto/tls, now with 100% more 1.3.

DO NOT USE THIS FOR THE SAKE OF EVERYTHING THAT'S GOOD AND JUST.

build passing
[dev.tls] crypto/tls: implement TLS 1.3 cipher suites
[dev.tls] crypto/tls: implement TLS 1.3 messages
[dev.tls] crypto/tls: implement TLS 1.3 record layer

https://go-review.googlesource.com/q/branch:+dev.tls
Deploying is hard

- First deployed Tris: draft 13
- Supported multiple drafts at a time ("hybrids")
- Browsers sometimes... diverged
### Build Jobs

<table>
<thead>
<tr>
<th>#</th>
<th>Job Status</th>
<th>Go</th>
<th>Client</th>
<th>Mode</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>206.1</td>
<td>✔️</td>
<td>1.7</td>
<td>interop</td>
<td>boring</td>
<td>5 min 26 sec</td>
</tr>
<tr>
<td>206.2</td>
<td>✔️</td>
<td>1.7</td>
<td>interop</td>
<td>bogo</td>
<td>4 min 9 sec</td>
</tr>
<tr>
<td>206.3</td>
<td>✔️</td>
<td>1.7</td>
<td>interop</td>
<td>tstclnt</td>
<td>6 min 49 sec</td>
</tr>
<tr>
<td>206.4</td>
<td>✔️</td>
<td>1.7</td>
<td>interop</td>
<td>picotls ZRTT=1</td>
<td>3 min 28 sec</td>
</tr>
<tr>
<td>206.5</td>
<td>✔️</td>
<td>1.7</td>
<td>interop</td>
<td>mint</td>
<td>2 min 57 sec</td>
</tr>
<tr>
<td>206.6</td>
<td>✔️</td>
<td>1.7</td>
<td>interop</td>
<td>gotest</td>
<td>3 min 3 sec</td>
</tr>
<tr>
<td>206.7</td>
<td>⚠️</td>
<td>1.7</td>
<td>interop</td>
<td>tstclnt ZRTT=1</td>
<td>6 min 21 sec</td>
</tr>
<tr>
<td>206.8</td>
<td>✔️</td>
<td>1.7</td>
<td>interop</td>
<td>boring REVISION=</td>
<td>5 min 17 sec</td>
</tr>
<tr>
<td>206.9</td>
<td>⚠️</td>
<td>1.7</td>
<td>interop</td>
<td>tstclnt REVISION=</td>
<td>6 min 56 sec</td>
</tr>
</tbody>
</table>
You may already be using it

• Firefox Nightly
• Chrome Beta (50%) / Canary

Secure Connection

The connection to this site is encrypted and authenticated using a strong protocol (TLS 1.3), a strong key exchange (X25519), and a strong cipher (AES_128_GCM).
Cloudflare Launch

Firefox Nightly

Chrome Field Test
https://tlswg.github.io/tls13-spec/
https://github.com.cloudflare/tls-tris
https://blog.cloudflare.com/tag/tls-1-3/

Nick Sullivan
@grittygrease

Filippo Valsorda
@FiloSottile
Y U NO ENCRYPT SNI!?
TLS 1.3 can’t encrypt SNI

Client Hello
- SNI
- Key share

No key negotiated yet

Server Hello
- Key share
- Certificate & signature
- Finished

Already has to pick certificate