Practical Crypto Attacks
Part 1

Emil Gurevitch

June, 2010

Outline

Outline

The General Idea

- A cookie authentication code (`auth_code`) is updated whenever the server changes a client's cookie.
- `auth_code` is generated as follows:

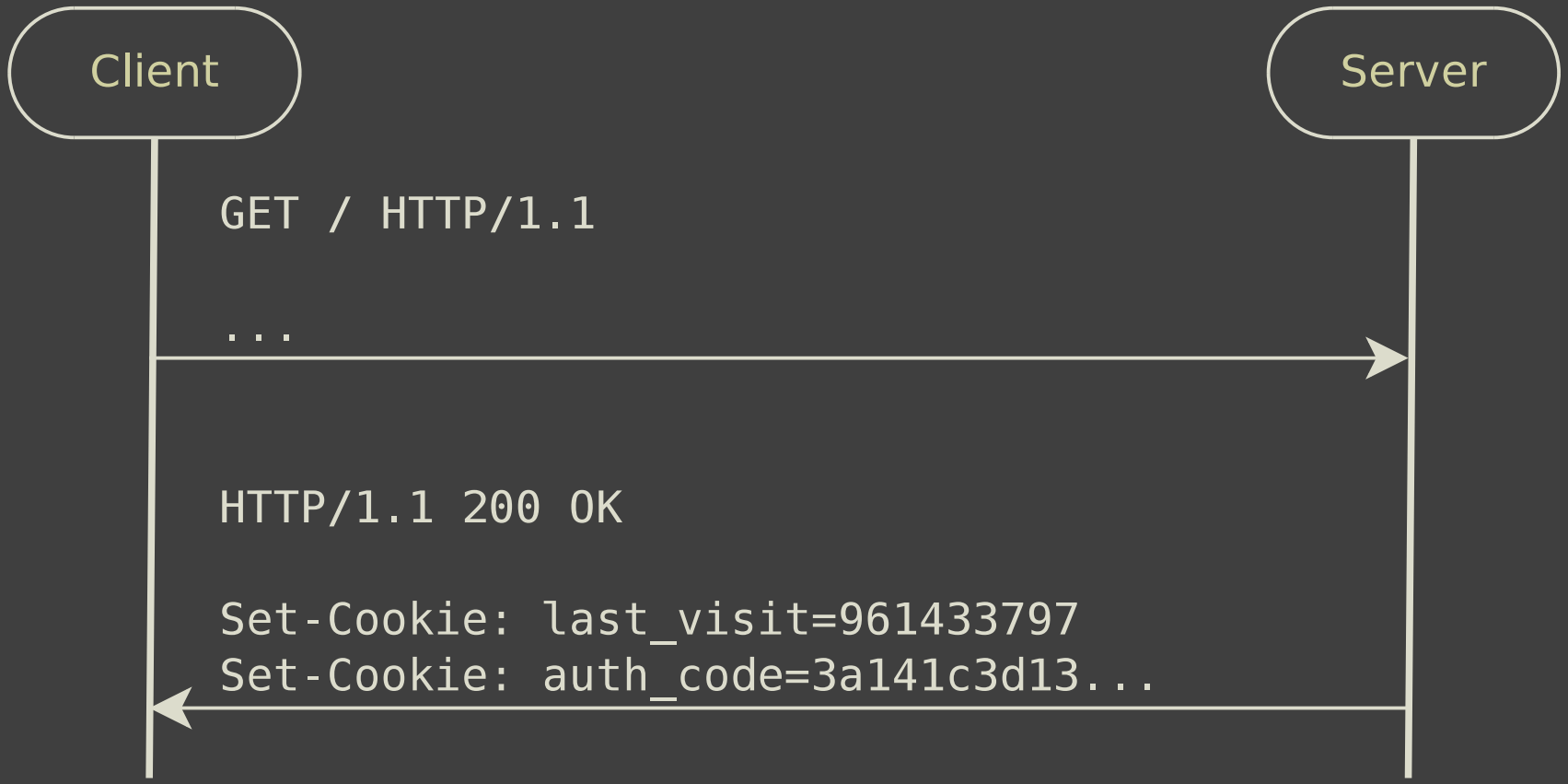
```
auth_code = sha512(secret-key + cookie)
```

- Once received by the client, `auth_code` is verified by the server as follows:

```
auth_code == sha512(secret-key + cookie)
```

- The attacker needs to know the `secret-key` in order to generate a valid `auth_code`, right?

Before Login



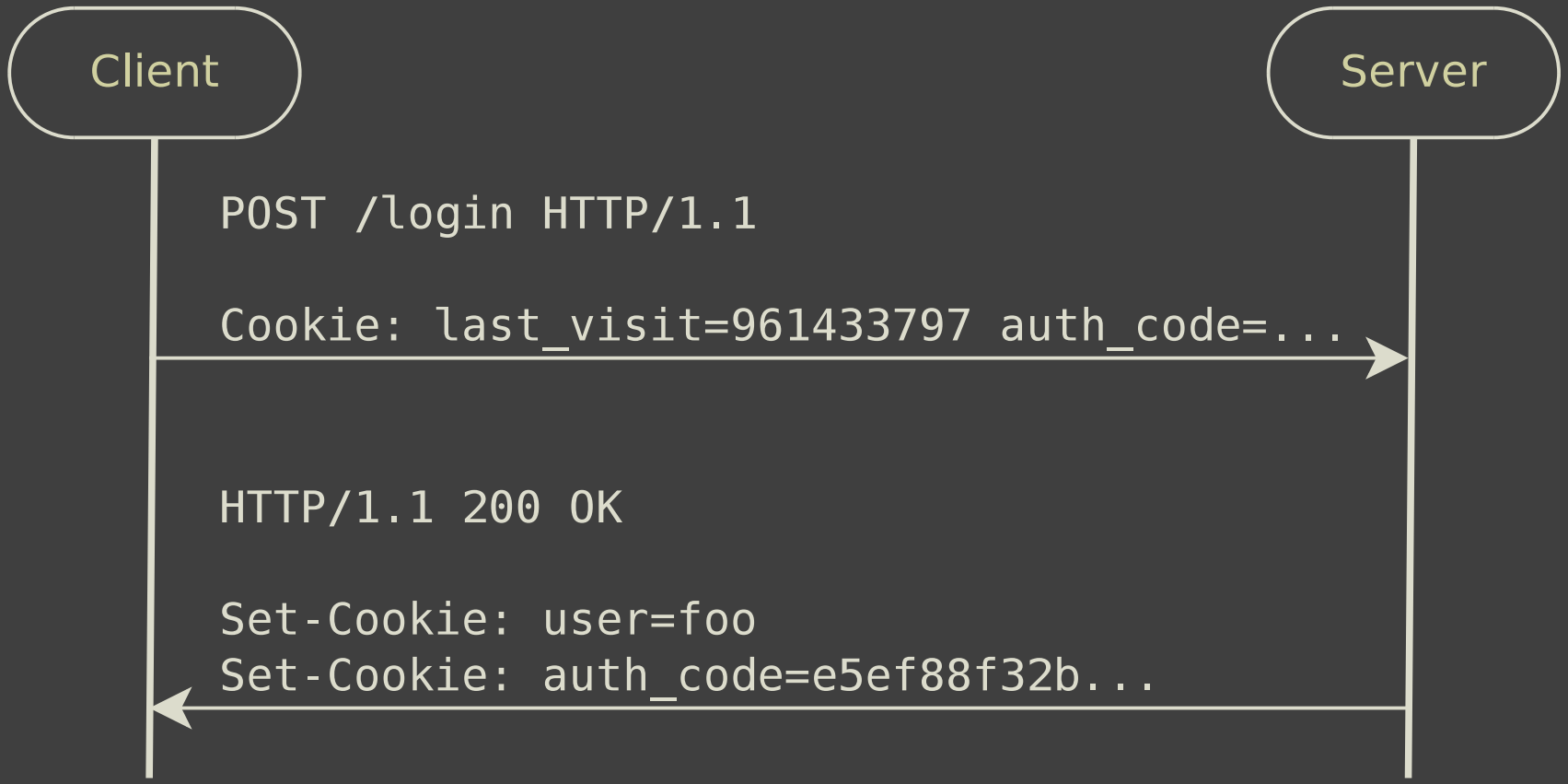
Client's cookie:

```
last_visit=961433797 auth_code=3a141c3d13...
```

Where:

```
auth_code = sha512(secret-key + "last_visit=961433797")
```

After Login



Client's cookie:

```
last_visit=961433797 user=foo auth_code=e5ef88f32b...
```

Where:

```
auth_code = sha512(secret-key + "last_visit=961433797 user=foo")
```

The Vulnerability

MD5, SHA-0, SHA-1 and SHA-2 families of hash functions all suffers from the length-extension bug:

Let

$$H = hash(m_1 + m_2 + m_3)$$

Then

$$hash(m_1 + m_2 + m_3 + m_4) = hash(H + m_4)$$

Assuming padding is corrected for.

The Attack

Add `user=admin` to the cookie (before login) and update the `auth_code` as follows:

```
auth_code = sha512(auth_code + "user=admin")
```

Assuming padding is corrected for, the attacker is logged in as `admin` (if user exists).

Lesson Learned

Do not use insecure hash constructions for authentication schemes, use a secure MAC function (such as HMAC-SHA512) instead.

Outline

The General Idea

- A webshop stores credit card numbers encrypted in a database.
- The encryption is done using the RC4 stream cipher with an über strong key.
- Without the key, an attacker is unable to get a hold of the credit card numbers, right? It depends. . .

Good Things To Know About RC4

- A keystream is generated that is XORed with the plaintext resulting in the ciphertext.
- The keystream is independent of the plaintext, it's derived from the key.
- A keystream must never be used more than once (true for most stream ciphers).

Known-Plaintext Attack

- Lets assume that
  - the attacker creates a profile on the webshop and submits a (bogus) card number (known-plaintext).
  - the attacker has access to the encrypted credit card numbers (ciphertext).
- The attacker can then mount a known-plaintext attack against the system.

Demo of Known-Plaintext Attack

Lesson Learned

Do not use the same keystream more than once when encrypting data using a stream cipher.

Outline

General Advice

- Do not develop your own crypto schemes.
- If you do, use high-level crypto APIs such as Keyczar and cryptlib.
- *If You're Typing The Letters A-E-S Into Your Code, You're Doing It Wrong* – Thomas Ptacek.

Questions/comments?