NameChain: A proof-of-concept implementation

Benedikt Kristinsson
Advisor:

February 2, 2016

Abstract

Traditionally we have come to rely on Certificate Authroities (CA) to verify that the servers we connect to are the ones we are relly trying to connect to. The CA system is very centralized and more based on beurocracy than technology.

In this essay we propose a distributed and peer-to-peer techonology that is builds a distributed consent of what encryption keys a client should expect a remote server to use. Rather than having to trust one central authority to tell the truth, the network forms a majority that is trusted.

1 Introduction

The current CA model is broken.

Operating systems and browsers come pre-loaded with CA certificates and will blindly trust their signatures. One rouge CA being distributed in such a way would make dragnet attacks almost trivial, and begs the question of wether they have been used for targetted attacks. In a post-Snowden world, we have to consider these to be facts, rather than mere speculations.

1.1 Contributions

The contributions of this work are the following:

- An implementation of a peer-to-peer network with Python and Twisted

2 Vendor-shipped CA certificates

The CA model instills trusts by having Certificates Authorities sign certificates. For the operating system or browser to be able to verify this signature, they need to maintain a list of known CA certificates, called *root certificates*. These lists come pre-shipped with the software and are thus hard to maintain and keep up-to-date.

If any of these root certificates have signed a certificate for any given domain, the software will accept the signature as valid and, more importantly - that the

remote server is who it claims to be. If a state-level actor has control over just one of the root certificates distributed (hereby referred to be as a *rouge certificate*) then they can man-in-the-middle the connection trivially. Occasionally, there have been doubts about the legitimacy of some of these certificates[1] and the Chinese government has at least one broadly distributed certificate.

Mozilla Project publishes their CA Certificate Policy [3] which details the application process, as well as how Mozilla maintains trust in the root certificates. More detailed discussion is mainted on the Mozilla Wiki [4], including lists of current CA certificates as well as a list of all removed CA certificates (although it does not detail the reasons for the removal).

3 Protocol description

NameChain is a peer-to-peer network to validate certificates with a majority consensus protocol. It is heavily influenced by the structure of Bitcoin [2]. As the proof-of-concept implemenation is not stable software and might be worked on after the finalization this essay, the protocol might change. Readers are directed towards the project on GitHub [1] for an up-to-date protocol description. Best effors are made to keep the `README.md` file for the project accuarte, but ultimately the protocol is specified by the sorce code.

4 Proof-of-concept implementation

The proof-of-concept implemenation is written in Python and uses Twisted[2] to implement the protocol and peer-to-peer network. The code is hosted and maintained on GitHub [1]. Messages are serialized to JSON-strings before being sent over the network, wrapped in an JSON envelope with a HMAC signature.

Listing 1: JSON envelope structure

```
def make_envelope(msgtype, msg, nodeid):
    msg['nodeid'] = nodeid
    msg['nonce'] =  nonce()
    sign = hmac.new(nodeid, json.dumps(msg))
    envelope = {'data': msg,
                'sign': sign.hexdigest(),
                'msgtype': msgtype}
    return json.dumps(envelope)
```

Seeing as this is a personal research PoC implementation, broad public use is not anticipated. The author currently maintains bootstrapping nodes at `freespace.sudo.is` and `ncnode.sudo.is`, with no guarantee of uptime.

[1] dig up examples
[2] `https://twistedmatrix.com/`

5 *

[1] Benedikt Kristinsson. benediktkr/ncpoc on github. https://github.com/benediktkr/ncpoc. Accessed: February 2, 2016.

[2] Satoshi Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008.

[3] Mozilla Project. Mozilla ca certificate policy. https://wiki.mozilla.org/CA:Overview. Accessed: Feb 2015.

[4] Mozilla Project. Mozilla ca program overview. https://wiki.mozilla.org/CA:Overview. Accessed: Feb 2015.