MUD (D)TLS profiles for IoT devices

draft-reddy-opswg-mud-tls-03

March 2019

T. Reddy (McAfee)

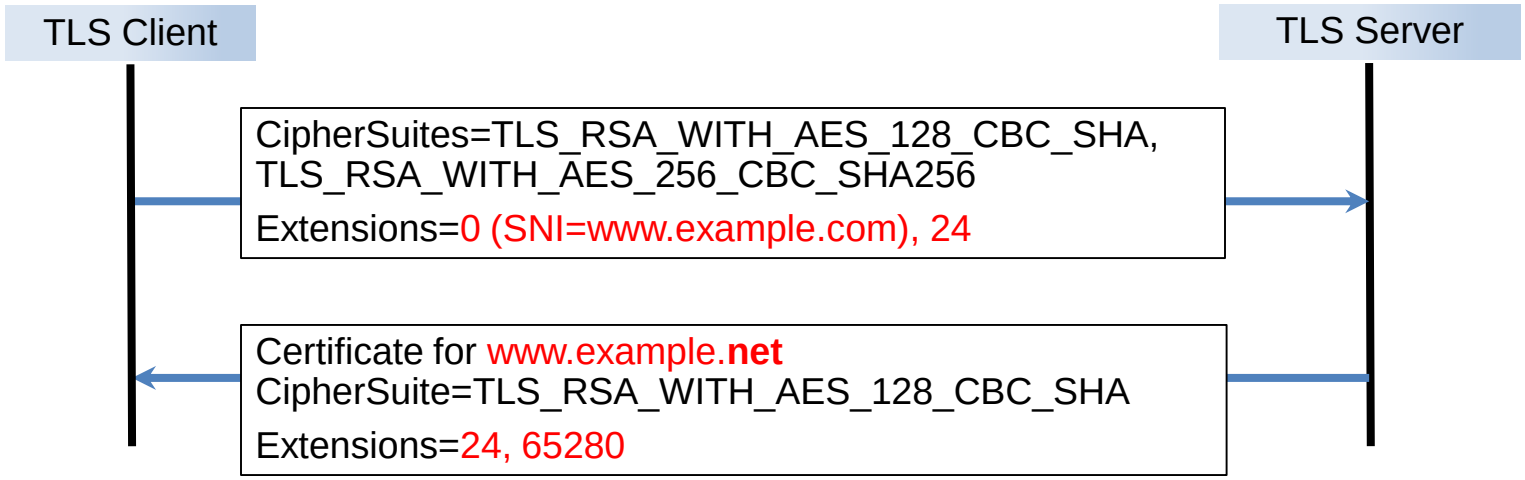
D.Wing (Citrix)

B.Anderson (Cisco)

MUD (D)TLS Goal

- The drafts proposes extending MUD to describe TLS interactions.

TLS handshake inspection



Malware TLS is different than legitimate software([1])

- SNI and SAN mismatch
- DGA pattern in SNI or SAN
- Offered/Selected Ciphersuites (ClientHello)
- Diversity of TLS extensions
- Self-signed

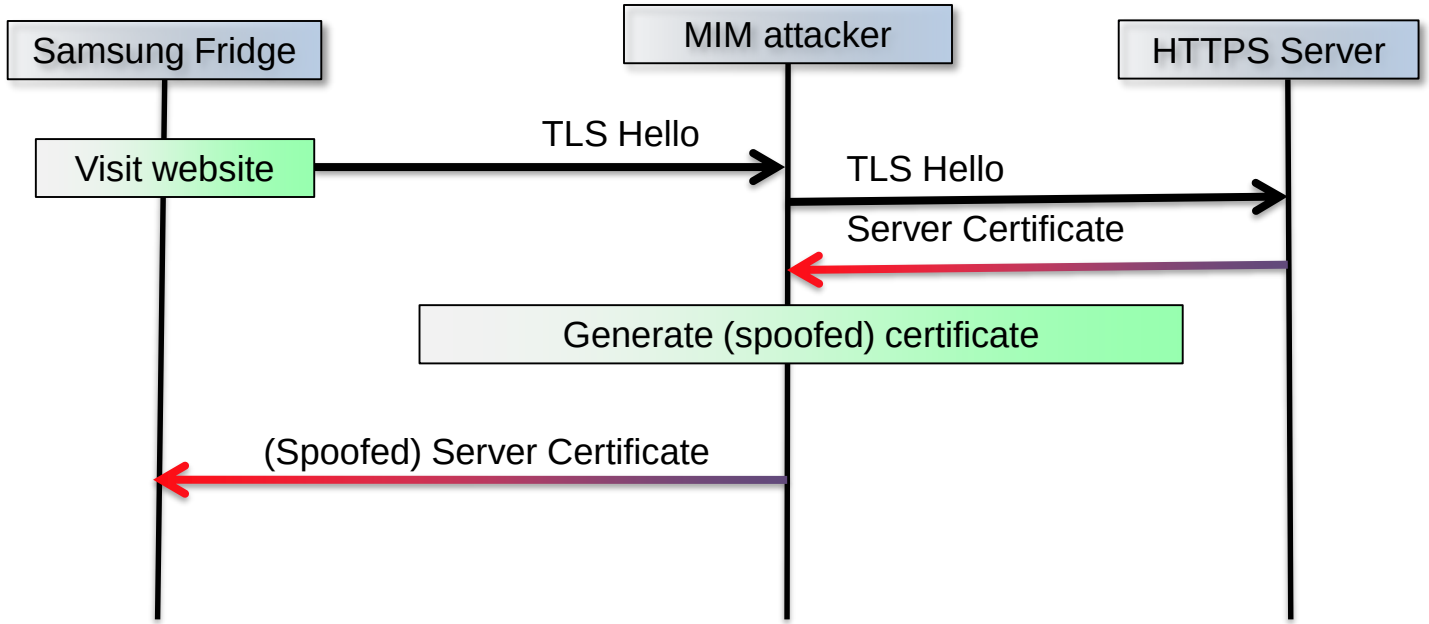
Detect broken TLS

- Best-practice failure (RFC7525)
  - Expired certificates
  - Poor-quality cipher suites
- Re-use of same private key ([2])
- Microsoft vulnerability to validate certificate ([3])

([1]) "Deciphering Malware's use of TLS (without Decryption)", https://arxiv.org/abs/1607.01639
([2]) "Millions of IoT Devices Using Same Hard-Coded CRYPTO Keys", http://thehackernews.com/2015/11/iot-device-crypto-keys.html
([3]) "Patch Critical Cryptographic Vulnerability in Microsoft Windows Clients and Servers", https://media.defense.gov/2020/Jan/14/2002234275/-1/-1/0/CSA-WINDOWS-10-CRYPT-LIB-20190114.PDF

Lack of certificate validation

- Samsung fridge failed to validate server certificate (see https://www.theregister.co.uk/2015/08/24/smart_fridge_security_fubar/)



Solution overview

- Benefits of MUD (D)TLS profiles for IoT devices include:
  - Ability to define policies for IoT devices that have diverse communication patterns
  - Robust against IoT devices learning new “skills” that change their communication patterns
  - Inadequate certificate validation by some IoT devices making them vulnerable to MiTM attacks

Observable (D)TLS profile parameters

- We profiled several IoT devices: Amazon Echo, Echo dot, Echo Show, Fire TV, Google Home Mini, Google Home and Kindle.
  - Observable (D)TLS profile parameters did not change after learning new skills. IoT devices have constrained TLS usage patterns.
  - (D)TLS profiles for IoT devices based on type, manufacturer and model is also different
- We also observed TLS profile parameters of thousands of malware flows.
- Growing trend of malware using TLS.

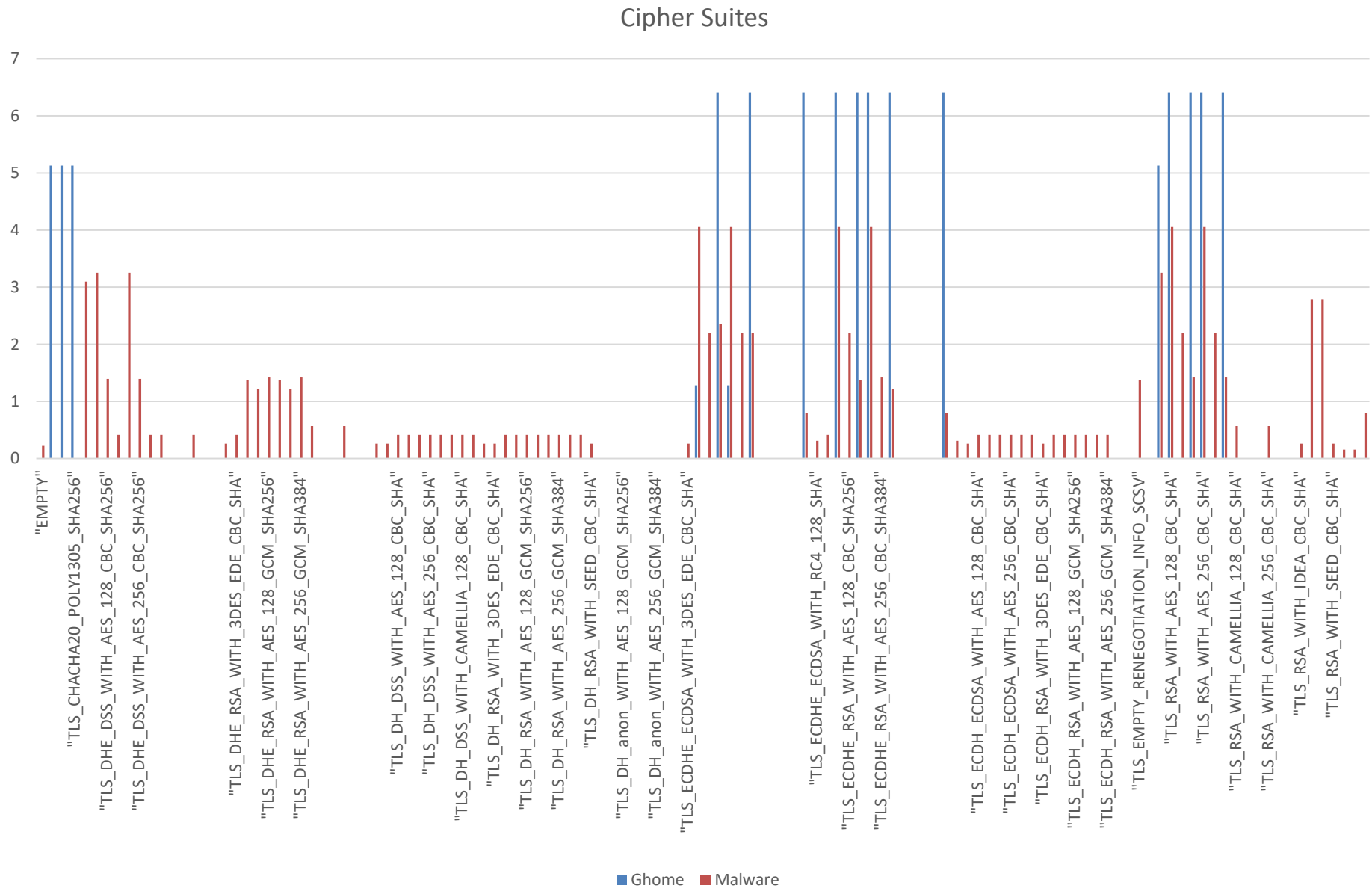
Malicious (D)TLS use can be blocked

Solution overview

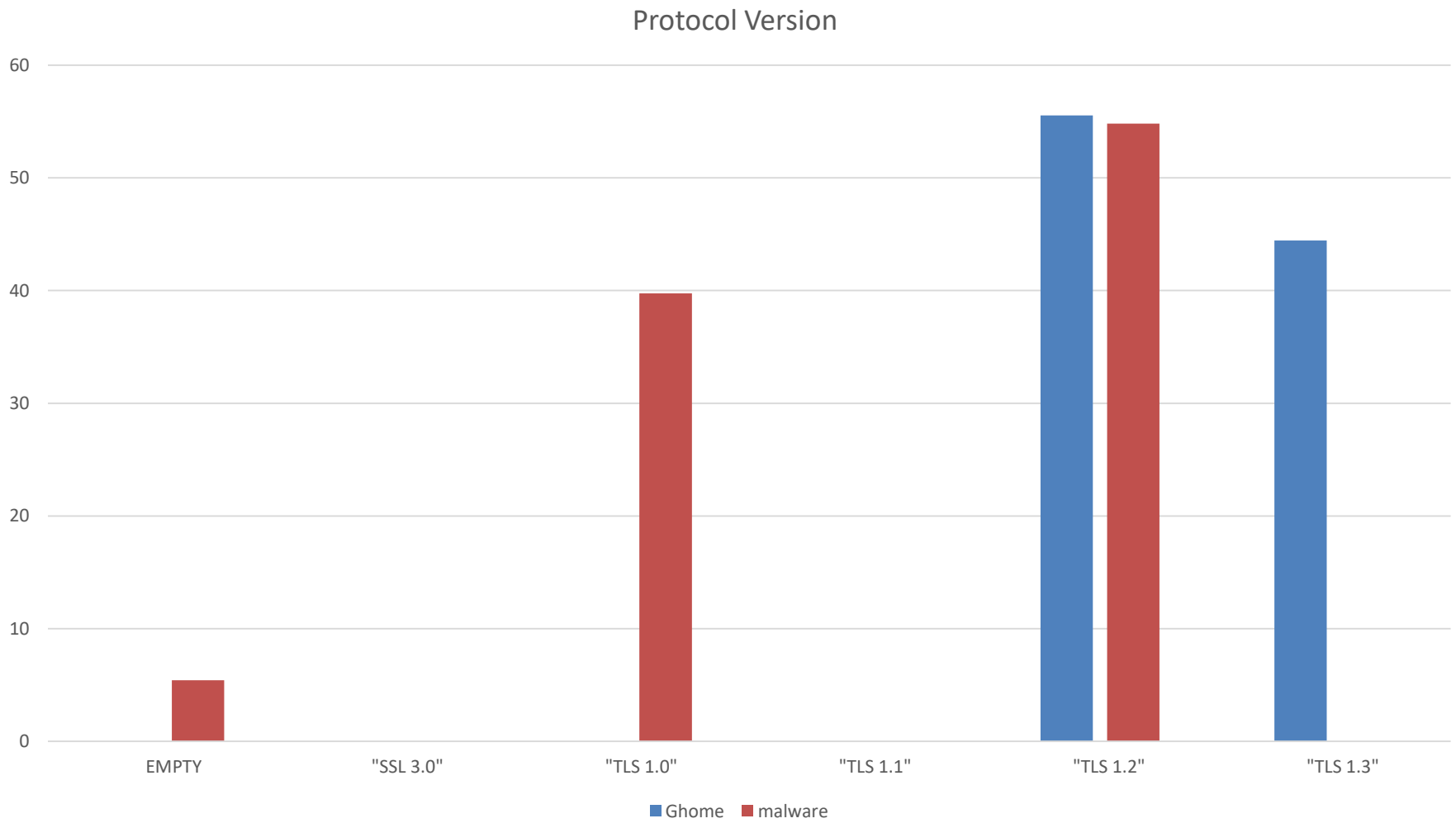
- Extends MUD to model observable (D)TLS profile parameters
- Client (D)TLS profile is defined once for re-use. (D)TLS profile for specific destination (e.g., Firmware server).

```
module: reddy-opsawg-mud-tls-profile
  augment /acl:acls/acl:acl/acl:aces/acl:ace/acl:matches:
    +--rw client-profile
       +--rw tls-profiles* [profile-name]
          +--rw profile-name                string
          +--rw protocol-version?           uint16
          +--rw supported_versions*         uint16
          +--rw grease_extension?           boolean
          +--rw encryption-algorithms*      encryption-algorithm
          +--rw compression-methods*        compression-method
          +--rw extension-types*            extension-type
          +--rw acceptlist-ta-certs*        ct:trust-anchor-cert-cms
          +--rw SPKI-pin-sets*              SPKI-pin-set
          +--rw SPKI-hash-algorithm?        iha:hash-algorithm-type
          +--rw psk-key-exchange-modes*     psk-key-exchange-mode
          +--rw supported-groups*           supported-group
          +--rw signature-algorithms*       signature-algorithm
          +--rw client-public-keys
             +--rw key-exchange-algorithms*     key-exchange-algorithm
             +--rw client-public-key-lengths*   client-public-key-length
```

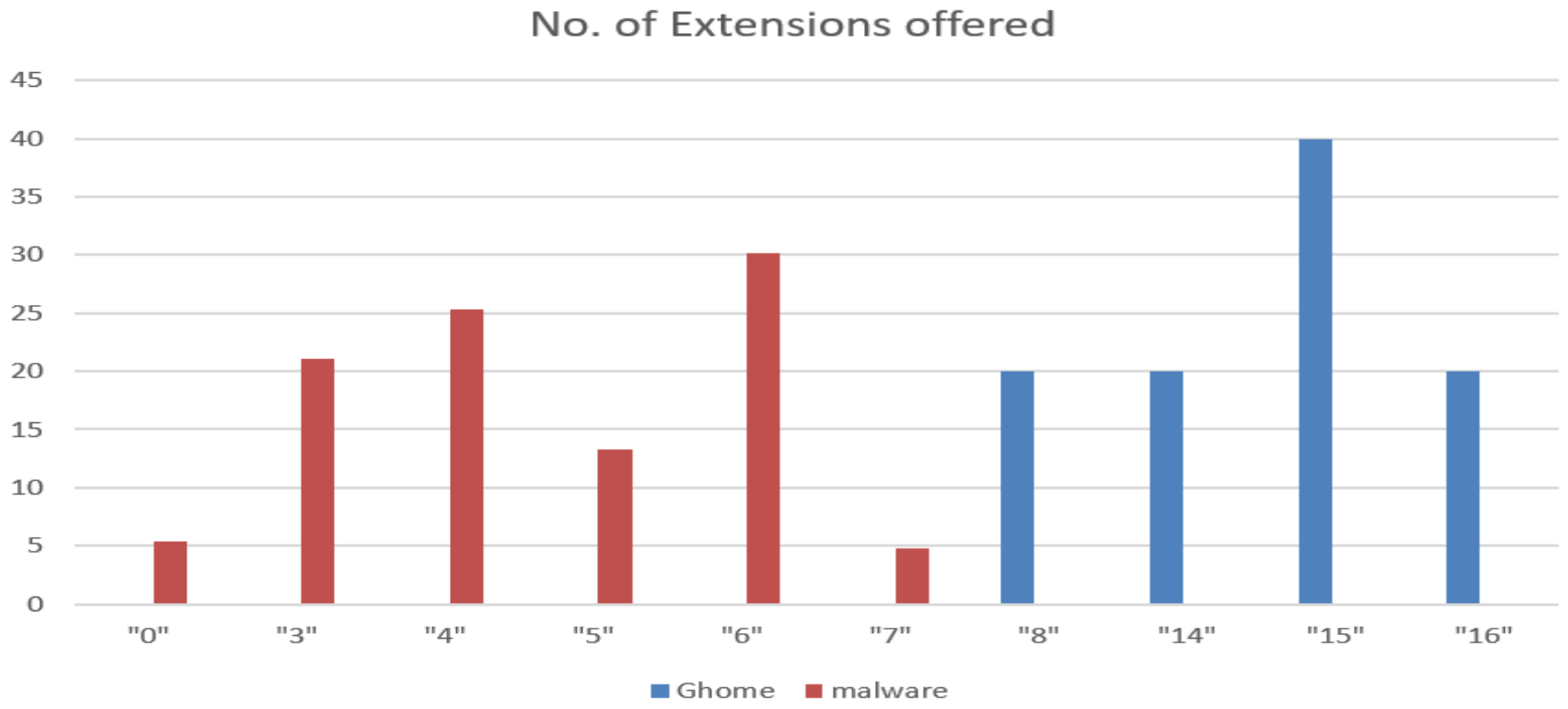
Google Home


Cipher Suites
7
6
5
4
3
2
1
0
"EMPTY"
"TLS_CHACHA20_POLY1305_SHA256"
"TLS_DHE_DSS_WITH_AES_128_CBC_SHA256"
"TLS_DHE_DSS_WITH_AES_256_CBC_SHA256"
"TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA"
"TLS_DHE_RSA_WITH_AES_128_GCM_SHA256"
"TLS_DHE_RSA_WITH_AES_256_GCM_SHA384"
"TLS_DH_DSS_WITH_AES_128_CBC_SHA"
"TLS_DH_DSS_WITH_AES_256_CBC_SHA"
"TLS_DH_DSS_WITH_CAMELLIA_128_CBC_SHA"
"TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA"
"TLS_DH_RSA_WITH_AES_128_GCM_SHA256"
"TLS_DH_RSA_WITH_AES_256_GCM_SHA384"
"TLS_DH_RSA_WITH_SEED_CBC_SHA"
"TLS_DH_anon_WITH_AES_128_GCM_SHA256"
"TLS_DH_anon_WITH_AES_256_GCM_SHA384"
"TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA"
"TLS_ECDHE_ECDSA_WITH_RC4_128_SHA"
"TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256"
"TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
"TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA"
"TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA"
"TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA"
"TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256"
"TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384"
"TLS_EMPTY_RENEGOTIATION_INFO_SCSV"
"TLS_RSA_WITH_AES_128_CBC_SHA"
"TLS_RSA_WITH_AES_256_CBC_SHA"
"TLS_RSA_WITH_CAMELLIA_128_CBC_SHA"
"TLS_RSA_WITH_CAMELLIA_256_CBC_SHA"
"TLS_RSA_WITH_IDEA_CBC_SHA"
"TLS_RSA_WITH_SEED_CBC_SHA"
Ghome
Malware

Google Home

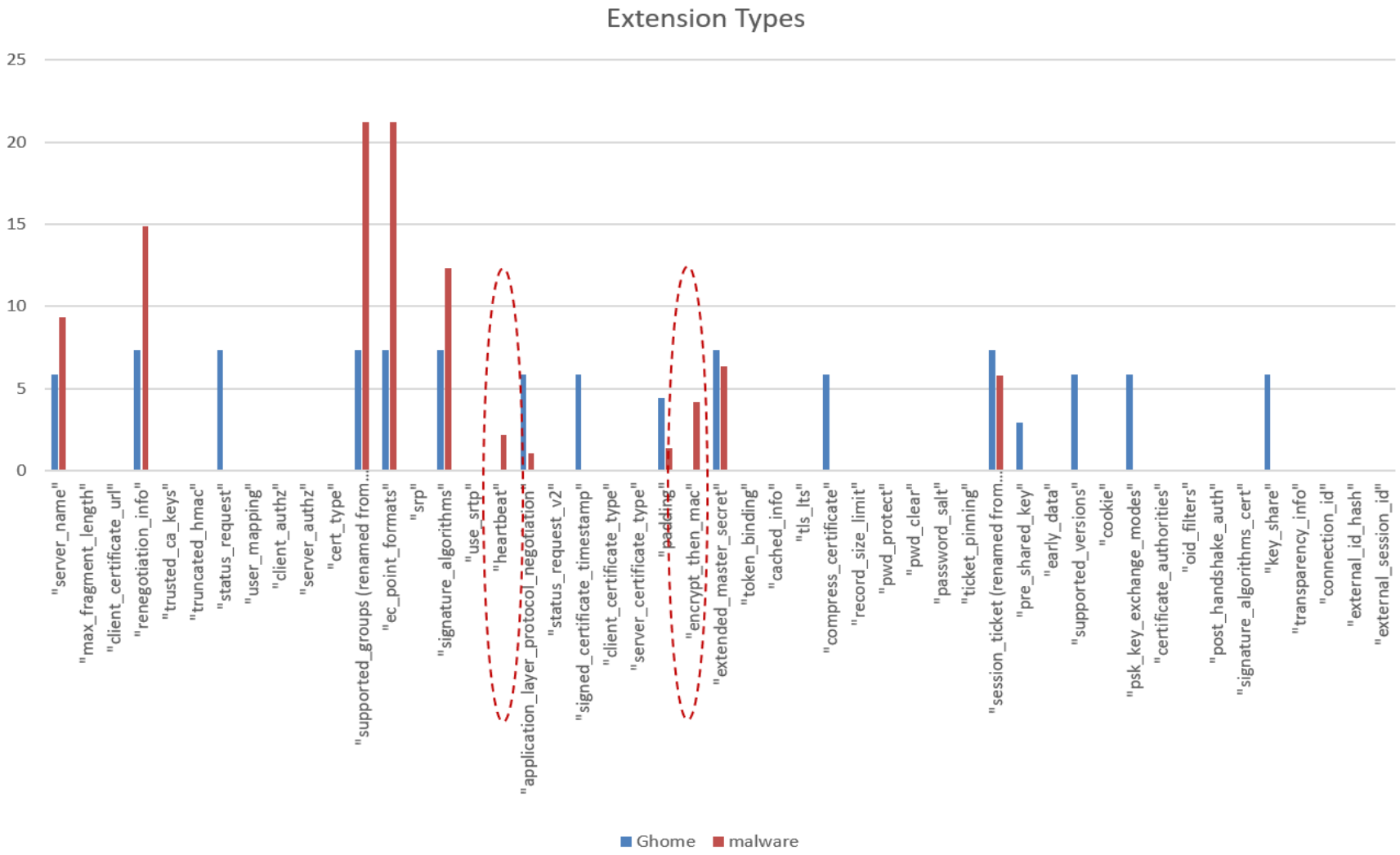

Protocol Version
60
50
40
30
20
10
0
EMPTY
"SSL 3.0"
"TLS 1.0"
"TLS 1.1"
"TLS 1.2"
"TLS 1.3"
Ghome
malware

Google Home

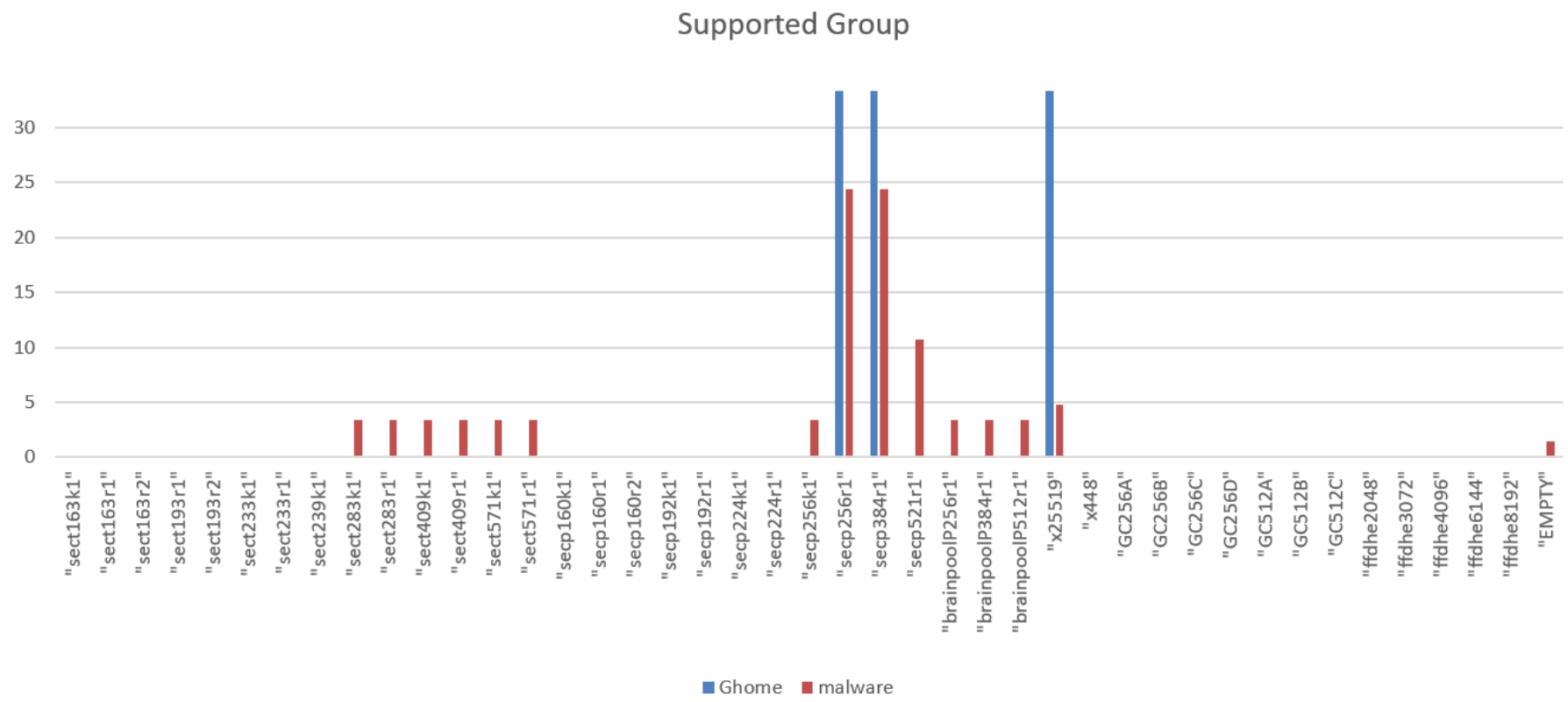


Malwares typically offer lesser number of extensions

Google Home

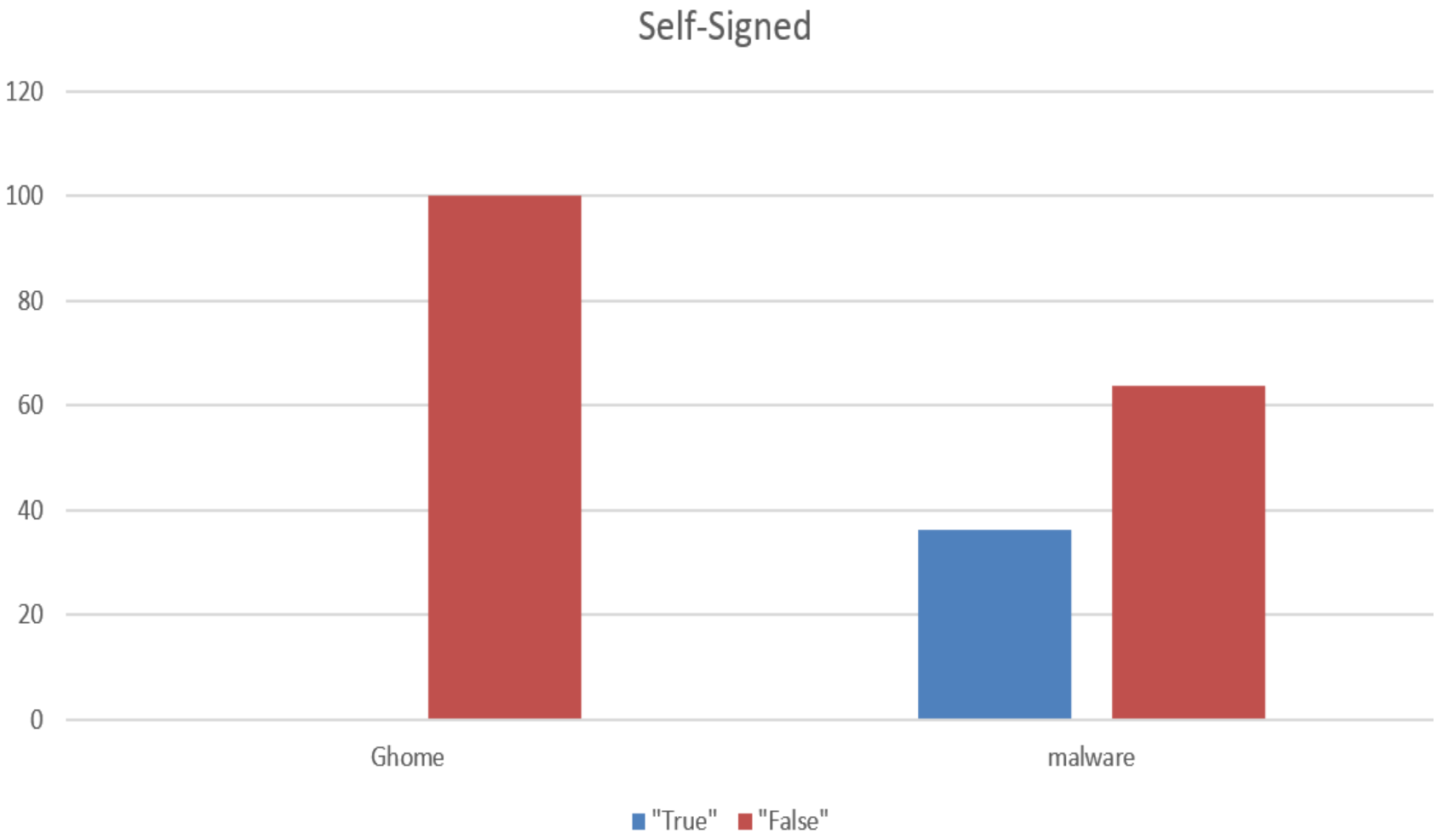

Extension Types
25
20
15
10
5
0
"server_name"
"max_fragment_length"
"client_certificate_url"
"renegotiation_info"
"trusted_ca_keys"
"truncated_hmac"
"status_request"
"user_mapping"
"client_authz"
"server_authz"
"cert_type"
"supported_groups (renamed from..."
"ec_point_formats"
"srp"
"signature_algorithms"
"use_srtp"
"heartbeat"
"application_layer_protocol_negotiation"
"status_request_v2"
"signed_certificate_timestamp"
"client_certificate_type"
"server_certificate_type"
"padding"
"encrypt_then_mac"
"extended_master_secret"
"token_binding"
"cached_info"
"tls_lts"
"compress_certificate"
"record_size_limit"
"pwd_protect"
"pwd_clear"
"password_salt"
"ticket_pinning"
"session_ticket (renamed from..."
"pre_shared_key"
"early_data"
"supported_versions"
"cookie"
"psk_key_exchange_modes"
"certificate_authorities"
"oid_filters"
"post_handshake_auth"
"signature_algorithms_cert"
"key_share"
"transparency_info"
"connection_id"
"external_id_hash"
"external_session_id"
Ghome
malware

Google Home



Malwares offer different supported groups

Google Home



Observable (D)TLS profile parameters

- Observed (D)TLS profile from several IoT devices and thousands of malware helped conclude intended (D)TLS use can be permitted and malicious (D)TLS can be blocked.
- Malware agents cannot mimic (D)TLS profiles of several IoT devices (type and model several manufacturers) and cannot keep up with the updates to (D)TLS profile.

TLS 1.3

- TLS 1.3 encrypts handshake but allowing inspection of several parameters:
  - List of cipher suites and extensions (e.g., supported versions, named groups, signature algorithms)
  - ServerHello chosen cipher
- Malware use of evasion techniques, such as ClientHello cipher suite randomization, can be detected.

TLS 1.3

- Full handshake inspection requires active participation in TLS 1.3:
  - Follow the behavior defined in Section 9.3 of RFC8446 to act as a compliant TLS proxy
  - TLS proxy for IT managed IoT devices
  - No need to inspect payload
  - Bypass acting as a proxy for connections destined to specific services due to privacy compliance requirements

draft-reddy-opsawg-mud-tls-03

- Comments and suggestions are welcome
- Collaboration to profile benign/malware flows on IoT devices