Mutual Declaration Mechanism of Multi-provider Relationship for Trusted Web Services

IETF116 Yokohama 2023/03/28

Wataru Ohgai, Takao Kondo, Korry Luke, Satoshi Kai, Keisuke Uehara, Satoru Tezuka
Keio University
alt@sfc.wide.ad.jp
What M2DMRT does?

M2DMRT
Mechanism of Mutual Declaration of Multi-provider Relationship for Trusted Web services

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Contribution</th>
<th>Approach</th>
</tr>
</thead>
</table>
| Declaration and verification of redirect relationship of multiple SPs | Light-weight, self-manageable declaration of trust using digital signature of opponent TLS public key | - Declaration of relationship using DNSSEC  
- Mutual declaration by related SPs |
TLS based security and redirection

- Security of Web is based on integrity assurance of hosts by TLS server certificate.
- Backend structure is being more complex, resulting in composing single Web service over multiple service providers (SPs) [5][6]
- This research focuses on the threat model regarding trust on multiple SPs’ relationship when redirecting service traffic.
Problem statement

1. The host integrity is not always the same as the service integrity.
2. TLS only assures the host integrity.

Integrity of single Web service containing multiple hosts/domain names
1. **Mutual** and **verifiable** declaration of service relationship
2. **Self-manageable** declaration of service relationship
3. **Minimum disclosure** of each party’s components

   Against the threat model

4. **Localization of transaction** of declaration modification
5. **Localization/minimization of failure points** (independent from central authority)

   System requirements
Requirements against feasibility

6. Adoption to the **Vertically-chained** Environment
7. Adoption Potential in **Horizontally-chained** Environments
8. Minimal **Processing Time** of Modification

Fig. 2.1: Example environment of horizontally-chained hosts.
Proposal: M2DMRT

```
pay-1.net._m2dmrt.ec-1.com. IN TXT (pay-1.net. ec-1.com. {K^pub_{ec-1.com.}} K^{priv_{ec-1.com.}})
ec-1.com. pay-1.net. {K^pub_{pay-1.net.}} K^{priv_{pay-1.net.}}
```

```
osok8r1kdet~~~.ec-1.com. IN NSEC3 ~~~
```

```
root zone
```

```
data part
```

```
com. zone
```

```
ec-1.com. zone
```

```
ec-2.com. zone
```

```
pay-1.net.
pay-2.net.
```

---

- Key Signing Key
- Zone Signing Key
- Declaration Record
- MRDA
- TLS Private Key
- Web server

<svg>
  <rect width="300" height="300" style="fill: #000000; opacity: 0.5"/>
  <text x="50" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">Key Signing Key</text>
  <text x="150" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">Zone Signing Key</text>
  <text x="250" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">Declaration Record</text>
  <text x="350" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">MRDA</text>
  <text x="450" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">TLS Private Key</text>
  <text x="550" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">Web server</text>
</svg>

<svg>
  <rect width="300" height="300" style="fill: #000000; opacity: 0.5"/>
  <text x="50" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">Sign</text>
  <text x="150" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">Hash</text>
  <text x="250" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">Communication</text>
</svg>

<svg>
  <rect width="300" height="300" style="fill: #000000; opacity: 0.5"/>
  <text x="50" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">DNS Records</text>
  <text x="150" y="250" style="font-size: 20px; fill: #000000; opacity: 0.5">Service Provider</text>
</svg>
Registration sequence (1)

MRDA: Mutual Relation Declaration Agent

Authenticates the opponent MRDA to register

EC site’s MRDA authenticates payment site’s MRDA

Payment site’s MRDA authenticates EC site’s MRDA

Generates shared key using DHE kex
→ Encrypts lasting communications with AES
Generates signature against payment site’s pubkey

Generates signature against EC site’s pubkey

Publish signatures as DNS records → Mutual declaration
Declaration record

DNSSEC signed
References


