Apache Milagro (incubating)

An Introduction
ApacheCon North America
Apache Milagro will establish a new independent security framework for the Internet
A Distributed Cryptosystem

Secure the Future of the Internet
Distributed Trust Ecosystem

- Milagro enabled apps and things receive their **key shares, or fractions**, from Distributed Trust Authorities.

- Keys have Identity "burned in"
Distributed Trust Authorities

- Anyone or organization can become a Distributed Trust Authority
- And run it in any geography or jurisdiction
- There is no PKI ‘root’ – the future is decentralized
Milagro Multi-Factor Authentication

Eliminates the risk of password database breach

Improves authentication / signature user experience

Improves authentication security to multi-factor

Identity based cryptographic multi-factor authentication and digital signature protocol that replaces passwords.

Milagro MFA runs entirely in software – it’s browser / app friendly.
Milagro TLS Library

Non-interactive: authenticate clients via a digital signature

Interactive: create certificate-less TLS with forward secrecy

The same protocol run interactively creates an authenticated key agreement between client & server or peer to peer
About Milagro Multifactor Authentication
A Toolkit for Multifactor Authentication

- Crypto Library (AMCL)
- MFA Server
- MFA JS Library
- MFA JS Client
- MFA Mobile SDK Core
- MFA Mobile SDK iOS
- MFA Mobile SDK Android
- MFA Mobile SDK Windows

MILAGRO
Zero-Knowledge Proof Authentication Without Passwords

Authentication into Web Applications

User registers in browser: receives a client secret and extracts a “PIN”

User logs in by using their PIN to recreate the client secret

Mobile Out-of-Band Authentication into Web Applications

User registers on mobile: receives a client secret on mobile device, extracts a “PIN”

User logs in by entering an access code from the website and PIN into the mobile app
Milagro MFA: How It Works

[Diagram showing the flow of authentication and secret sharing between customer, application, RPS, Auth. Server, D-TA Proxy, and D-TA 1 and D-TA 2.]
Milagro MFA: Registration
Milagro MFA: Authentication

Application

RPS

Auth. Server

D-TA 1

F(num, key)

num

Auth. request

D-TA 2

num

F(num, key)
Milagro MFA Integration
Let The Machines Work For Human Comfort

- At least 12 characters from upper-case and lower-case letters, and ...

- You must change it every 2 month.

- You must choose independently random passwords for all accounts.

- 4 digit number is OK for PIN. Resiliency against brute force attacks.

- You do not need to change secrets. Zero-knowledge proof without credential database, hence no breach.

- You may use the same PIN for all accounts. Machine generates random OTP from the two factors, with your identity burned in.
Override the standard password login with Milagro-MFA without modifying the code.
Milagro: Plug-In Architecture

MPIN.js produces digitally signed one time token using PrivateKey, and submit it to the password form.

- PIN (Human part)
- PrivateKey – PIN (Machine part)
- Timestamp

Authentication Proxy intercepts bind requests and verifies the signature.
Milagro: Implementation

MPIN.js communicates with MPIN server to submit full message of signed token. MPIN.js submits tokenized message (typically hash value) in the password form.

MPIN.js communicates with MPIN server to submit full message of signed token. MPIN.js submits tokenized message (typically hash value) in the password form.
Milagro: Easy Installation

MPIN.js overrides the standard password login form.
Milagro: Easy Installation

1. Import MPIN.js at the frontend.

```html
<link href="https://public.milagro.io/public/css/mpin.min.css" rel="stylesheet">
<script src="https://public.milagro.io/public/js/mpin.js"></script>
```

2. Insert LDAP proxy from Milagro between your target application and the LDAP server.
About Milagro TLS Library
Certificate Less TLS with Perfect Forward Secrecy

An extension to the ARM mbed TLS library (https://tls.mbed.org/)

Introduces two new key-exchange algorithms

Designed for Client-to-Server communication (MILAGRO_CS)

Designed for Peer-to-Peer communications (MILAGRO_P2P)
## New Environments Need New Solutions

<table>
<thead>
<tr>
<th>Current Convention</th>
<th>Milagro TLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open SSL Complexity</td>
<td>Easily Auditable</td>
</tr>
<tr>
<td>Certificate Management Complexity in TLS</td>
<td>X-509 Digital Certificates Not Needed</td>
</tr>
<tr>
<td>Certificate / Key Revocation</td>
<td>Revocation That Works (time and identity based)</td>
</tr>
<tr>
<td>Limited Security for IoT &amp; Containers</td>
<td>Identity based C/S and P2P Channel Security</td>
</tr>
</tbody>
</table>
Standard TLS

CLIENT

ServerHello

ServerKeyExchange

ServerHelloDone

ClientKeyExchange

ChangeCipherSpec

ClientFinished

ChangeCipherSpec

SeverFinished

SERVER

ExchangeData

CLIENT

SERVER
WARNING!!
MAY CONTAIN COMPLEX MATH PROBLEMS, WHICH CAN CAUSE HEADACHE AND SERIOUS CONFUSION...
CLIENT AUTHENTICATION – step 1
We are getting: time value, generate random number, we hashed Client identity etc.
(even more complex MATH… 😊)

Generating AES Key

ClientHello + \{A,U,V,t\}

ServerHello

ServerKeyExchange = \{W\}

ServerHelloDone

ClientKeyExchange = \{R\}

ChangeCipherSpec

ClientFinished

ChangeCipherSpec

ServerFinished

ExchangeData – identity

V_{G1} = (x+y)_G_{G1}
What Milagro TLS Can Deliver

- Data Center Cryptosystem
- Orchestration Host Security
- Container 2 Container
- Websites and TLS w/o Certs
- MQTT & CoAP

- Distributed Trust
- Pairing Based Cryptography (IBE)
- Easy Revocation Through Invalidation
- Certificate Authority or self signed certs not needed
- Enables P2P TLS for devices and containers
Pre-Alpha Code Available Now

Milagro – DTA code
Pre-Alpha: Extended mbedtls library
Pre-Alpha: P2P Wang / Chow-Choo library
Draft Milagro TLS White Paper
Demo at booth on CS & P2P

WARNING: May Contain Nuts
What’s Ahead
Developing Milagro (draft)

Q2'16
- Milagro MFA 1.0
- Milagro TLS 0.1

Q3'16
- Milagro TLS 1.0

Q4'16
- Distributed Cryptosystem for Datacenters 0.1

Q1'17
- Distributed Cryptosystem for Datacenters 1.0