Title: Dynamic Distributed Key Infrastructures (DDKI) and Dynamic Identity Verification and Authentication (DIVA)

Author: André Brisson

Summary: Dynamic Identity Verification and Authentication (DIVA) is a one-time-pad protocol for continuous, dynamic authentication throughout a session.

Dynamic Distributed Key Infrastructures is a distributed, virtual secure framework comprised of components and devices enabled with DIVA. These frameworks require only the single, initial, secure key distribution.
Dynamic Identity Verification and Authentication (DIVA) is a One-Time-Pad

Both server and endpoint have a copy of the account identity management key. The server sends a request to the endpoint for an identification token of a specific length – in this case, 25 bytes. It is not sending across either an offset or a key with this request.

Device state 1a

Last valid offset

22 1F CB FE FA 17 F2 8E A5 F0 8A E1 55 D6 DD 36 13 73 E2 9A 65 2F F6 EA 71 FE F7 D7 B8 26 5D 26 8B 93 64 16 03

The key stream is a minimum of $10^{60}$ bytes in length. We are continuously and dynamically comparing tokens to ensure the correct identity of the network user. A token is an unused segment of key stream of an arbitrary length. It is random and has the equivalency of being encrypted – it cannot be guessed or broken, and it is only used once.

The endpoint replies by sending a 25-byte token, beginning at its last valid offset.

Device state 1b

Last valid offset plus token

22 1F CB FE FA 17 F2 8E A5 F0 8A E1 55 D6 DD 36 13 73 E2 9A 65 2F F6 EA 71 FE F7 D7 B8 28 5D 26 8B 93 64 16 03

Length = 25 bytes. This is arbitrary and scalable depending on security requirements.
DIVA One-Time-Pad Synchronization

DIVA dynamic update of offset

Server authenticates user/device by comparing the received token bit-by-bit to the token generated at the Server for this account/person/device. If they are identical, then:

- Server acknowledges by sending authorization
- Both server and endpoint update dynamic offset independently

New offset = last offset + token + 1

Last offset

```
22 1F CB FE FA 17 F2 8E A5 F0 8A E1 55 D6 DD 36 13 73 E2 9A 65 2F F6 EA 71 FE F7 D7 B8 28 5D 26 8B 93 64 16 03
```

Length = 25 bytes. This is arbitrary and scalable depending on security requirements.

The system is synchronized for the next continuous authentication query.

The account is automatically locked if the comparison of tokens fails. This would happen if someone has copied a key and the offsets are not synchronous.
Inherent Intrusion Detection

100% accurate – only two DIVA outcomes

Someone tries to steal a key.

1. The legitimate user logs back onto the network first.

   - The legitimate key and server offset dynamically updates with this use independently.
   - The pirated or spoofed key (if possible) is no longer synchronized with the server and the legitimate key.
   - The pirate will be detected if he makes a login attempt.
   - The pirate can’t access the network. Stolen copy is useless.
   - No theft has occurred.

This is the only outcome we have ever seen.
2. The pirate somehow steals a key and logs on first.

   - The offset at the server and pirated key updates with this use.
   - The legitimate key is no longer synchronized with the server.
   - The next time the legitimate owner logs on to the secure network, the server recognizes that the offset is no longer synchronized because of the pirated key.
   - The account is automatically locked.
   - System Administration and client know that their account has been accessed.
   - The logs know the exact duration of the event and the exact transactions within that time, beginning at the last time the server and client were synchronized, and ending at the point in time when the account was locked.
   - The pirate IP address is known for law enforcement use.
Whitenoise prevents all known and anticipated cyber attack classes

• **Man-in-the-Middle attacks** are prevented because there is no key or offset exchange during a session. In a distributed key system the key is securely delivered one time. After, in session, **there is no key transfer with DIVA**.

• **Side Channel attacks** are prevented because all operations are order 1 after key load and because there is no access to the master key. Proper chip design has Whitenoise operating as a one-time-pad. The attacker has no access to the key refreshing the circuits. Two additional techniques were presented by the University of Victoria, British Columbia. They will be shown in afternoon sessions by Dr. Mihai Sima.

• **Mathematical and factoring attacks** are not possible because keys are created by a binary, mechanical process more akin to Enigma and not an arithmetic or mathematical process.

• **Botnet attacks** are prevented by configuration with server so the botnet never has access to all the key material needed to authenticate data being sent OUT of a network or computer.

• **Brute force attacks** are not feasible with the continually changing dynamic offsets and exponential key lengths.

• **Denial of service attacks** can be prevented by exploiting unbreakable identity and a proxy for secure network access so that hackers could never get on a network.
Title: Dynamic Distributed Key Infrastructures (DDKI) and Dynamic Identity Verification and Authentication (DIVA)

Author: André Brisson

Summary: Dynamic Distributed Key Infrastructures is a distributed, virtual secure framework comprised of components and devices enabled with DIVA. These frameworks require only the single, initial, secure key distribution.

Dynamic Identity Verification and Authentication (DIVA) is a one-time-pad protocol for continuous, dynamic authentication throughout a session.
Here’s how — with one unbreakable key

- Identify network and data access of all users, components and administrators with unbreakable, dynamic, continuous, one-time pad authentication
- Log all network activity
- Unique authenticated one-time pad encryption for all persons, components and data
- Keys are always encrypted
- Stored private keys are separate from their encrypted offsets (with Whitenoise)
- Internal access to keys or offsets require two different sets of administrators
  
  *If criminals could manage to get in your networks, they need to access key structure and offsets that are located in two separate places and encrypted with different keys simultaneously and use them before the next authentication call.*
- Transmissions and communications are always encrypted
There are 3 main components and 2 framework choices for crypto systems

Crypto systems

• Key creation
• Key distribution
• Key management

There are only two options in security frameworks

• Public key, asymmetric frameworks (broken badly)
• Dynamic Distributed Key Infrastructure (DDKI) frameworks (never broken)

With only two choices, DDKI is analogous to Tesla using alternating current and Edison using direct current.
### The PKI Inefficient Handshake

Network security and performance are exasperated by computationally intensive handshakes to initiate sessions.

<table>
<thead>
<tr>
<th>Alice</th>
<th>“Evil” Eve</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice and Bob exchange a Prime (P) and a Generator (G) in clear text, such that P &gt; G and G is Primitive Root of P. G = 7, P = 11.</td>
<td>Evil Eve sees G = 7, P = 11</td>
<td>Alice and Bob exchange a Prime (P) and a Generator (G) in clear text, such that P &gt; G and G is Primitive Root of P. G = 7, P = 11.</td>
</tr>
<tr>
<td>Alice generates a random number: $X_A$. $X_A = 6$ (Secret)</td>
<td>Bob generates a random number: $X_B$. $X_B = 6$ (Secret)</td>
<td></td>
</tr>
</tbody>
</table>

- **Encryption:**
  - $Y_A = G^{X_A} \pmod{P}$
  - $Y_A = 7^6 \pmod{11}$
  - $Y_A = 4$
  - $Y_B = G^{X_B} \pmod{P}$
  - $Y_B = 7^6 \pmod{11}$
  - $Y_B = 8$

- **Key Exchange:**
  - Alice receives $Y_B = 8$ in clear text.
  - Evil Eve sees $Y_A = 4$, $Y_B = 8$.
  - Bob receives $Y_A = 4$ in clear text.

- **Secret Key:**
  - Secret Key = $Y_B^{X_A} \pmod{P}$
  - Secret Key = $8^6 \pmod{11}$
  - Secret Key = 3
  - Secret Key = $Y_A^{X_B} \pmod{P}$
  - Secret Key = $4^6 \pmod{11}$
  - Secret Key = 3
In symmetric, dynamic, distributed key systems, the server has copies of all the keys on a system. The keys are stored in an encrypted state. The keys are always kept separate from the last current dynamic offsets.

Each endpoint has only its unique, distributed, private/secret key. Secret keys are NEVER shared between endpoints. There is never key or offset exchange after setup. The following illustration shows a system in its simplest configuration.
The simple DDKI session key handshake

**Server has copies of all keys on system:**
- pkA
- pkB
- etc.

**Whitenoise DIVA Dynamic Distributed Key Authentication Server**

2. The server creates a session key with its master key and gives it a session key identifier. It locates A's unique, private key and authenticates A. The server encrypts the session key and sends the key and identifier back to A. **The private secret key is never shared with another endpoint.** The only arithmetic function used is X-Or, the fastest function on a computer.

5. The session key is encrypted with pkB and sent back to “B”.

**Server has unique master key to identify itself and make keys. It has link keys to other servers for secure communications at the server level**

1. Source “A” with unique private key
2. pkA requests link with “B”.
3. “A” decrypts the session key with its pkA, encrypts the message with the session key, and sends encrypted message with the session key identifier to “B”.
4. Destination “B” with unique private key
   - pkB requests session key
   - pkB requests session key and sends identifier.
5. “B” decrypts session key with pkB and then decrypts message.
Dynamic Distributed Key systems fix your networks

➢ Key creation
  • Your private master key makes an unlimited number of unique, private dynamic, one-time-pad keys with exponential length and strength

➢ Key distribution
  • Master keys are securely delivered with preconfigured servers
  • You, in turn, create your own keys and distribute them one-time to your own network endpoints

➢ Key management
  • You are the only one that has copies of all the keys on your network
  • Keys and offsets are encrypted with separate keys and kept in separate places

  • Data is stored with one-time-pad authenticated encryption that imposes provenance and identifies data ownership
  • Identity is guaranteed with unique, unbreakable keys and dynamic, continuous, one-time pad authentication
  • All network usage is logged
Hybrid – Distributed OTP PKI – a simple extension

• Adding **Dynamic Identity Verification and Authentication** (DIVA) to your PKI framework creates a two channel (both asymmetric and symmetric calls), multifactor secure network access challenge.

• Hackers have to break two keys simultaneously, one is a continuous-dynamic-one-time-pad, for each and every break attempt.

Technical validations

- Telecom Council of Silicon Valley (TCSV) 2014 Nokia Global Cloud and Colossal Data Innovation Challenge
- TCSV Showcase 2015 Invitee
- TCSV Spiffy San Andreas award 2015 finalist
- United Nations International Telecommunications Union presentation
- European Telecommunications Standards Institute
- White House Invitation to the First National Cyber Leap Year Summit
- Director of Global Cyber Security DHS – vetted Whistnoise with US government before becoming CEO temporarily
- AT&T Certification
- Booz Allen Hamilton Innovation center
- Booz Allen Hamilton Common Criteria Labs
- University of California Berkeley security analysis and validation – National Research Council of Canada
- University of Victoria, British Columbia performance analysis – National Research Council of Canada
- Tata Innovation Labs
- Sprint Advanced Technology Labs
- MIT Entrepreneur accelerator
- Canadian Technology Accelerators Boston and India
- Raytheon IPSEC Future of Security Finalist
- Global Security Challenge Finalist 2010
- Global Security Challenge Finalist 2013
- Global Black Hat Security Challenge – no winner
- Global DEFNCON Security Challenge – no winner
- Global Whistnoise Challenge – no winner
- Smart Grid British Columbia Institute of Technology
- BCIT CTA Advisory Board
- Sierra Wireless Best in Class Award
- ISO/IEC JTC SC27 expert member – many groups

http://www.wnlabs.com or the technical website http://www.wnlabs.com/home.php for all papers, research, awards and notices.