DIARY

JANUARY 2015

M	T	W	T	F	SA	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

« dec

Breakfasts FIC days Partner events

NEXT EVENT

There is no further event for the moment.

LAST PUBLICATION

[Policy paper] Cybercrime and social networks: Dangerous liaisons

FIC ACTUALITY

Executive Strategic Simulation by Thales 15/01/2015

Bruce Schneier's lecture will take place on Wednesday 21st of January, 2:15 pm 13/01/2015

The British National Cyber Crime Unit will be present at the FIC 2015 12/01/2015

The United States: a stronger presence at FIC 2015 12/01/2015

NATO: general Paloméros, Supreme Allied Commander of Digital Transformation, will speak at the FIC 2015 12/01/2015

RECEIVE OUR NEWSLETTER

Name

Email

Message

Send

OPERATIONAL MANAGEMENT OF SECURITY ISSUES

Articles - 21/11/2014

Dynamic Identity Verification and Authentication (DIVA) [by André Jacques Brisson, Whitenoise Laboratories Canada Inc.]

Publié par André Jacques Brisson

ABSTRACT

Dynamic Identity Verification and Authentication (DIVA) secure communications for all networks through the cloud and create secure virtual networks within the cloud.
DIVA is a secure protocol that prevents all cyber security and identity theft attacks by demanding proof of identity at the time of network access and throughout the network session. It assigns provenance to all data.
A network user is provided a unique, unbreakable, one-time-pad identity.
The identity proof calls occur at a rate faster than is possible to breach the network. The identity is constantly changing and the hacker is constantly starting over on an unsolvable challenge.
Dynamic Distributed Key Infrastructures (DDKI) are secure virtual network-of-networks of persons and devices that use Dynamic Identity Verification and Authentication.
International standards organizations have articulated a defined need for large, dynamically authenticated, distributed platforms and services AND large, distributed, on-line authentication systems where there is only partial disclosure of credentials. These are requirements necessary for: secure cloud computing, securing critical infrastructures and secure identity based telecommunications.
Network security cannot be achieved in any context without proper identity management of all network endpoints (persons and devices) and of all components comprising the telecommunications backbone

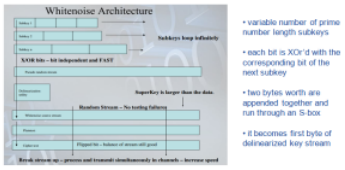
INTRODUCTION

We will first look at how a Whitenoise key is constructed. We will then look at its one-way functions. Finally, we will then look at how Dynamic Identity Verification and Authentication exploit unique Whitenoise keys, how DIVA works and why it operates as a one-time-pad.

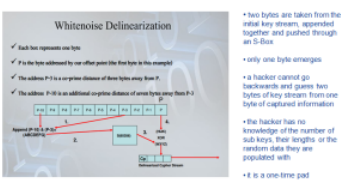
CREATING A WHITENOISE KEY

One distributed Whitenoise key creates an infinite number of one-time-pads and handles all network security controls. Whitenoise is a deterministic random number generator creating deterministic key streams of unlimited length that are orders of magnitude more random then radio-active decay.

A Whitenoise key is built from a variable number of prime number length sub-keys which roll out horizontally to create the data source. Each bit from the data source is XOr'd with the corresponding bit of the next subkey in a vertical fashion to create the first key stream. This ensures that it is not operating like a line feed shift register or a counter and makes it bit independent.



To delinearize this stream, two bytes worth from the initial key stream are appended together and run through an S-box. Only one byte emerges. That becomes first byte of the delinearized key stream which can then be used for any cryptographic purpose.
This creates several one-way functions. A hacker cannot go backwards and guess two bytes of key stream from one byte of captured information. The hacker has no knowledge of the number of subkeys in the data source, their lengths or the random data they are populated with. Further, it is used as a one-time pad in the DIVA protocol and a one-time pad is the only mathematically proven unbreakable key technology.



One-time-pads have three characteristics:

OUR CONTRIBUTORS

Julien Nocetti

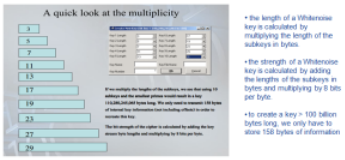
PHOTO GALLERY

TWITTER

One-time-pads have three characteristics:

1. The keys are larger then the data to be encrypted or monitored.
2. The keys are random.
3. The keys are never used more than one time.

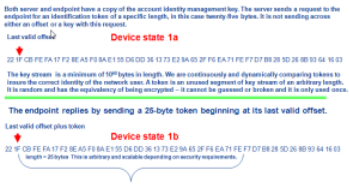
David Wagner of the University of California, Berkeley did a security analysis of a deployment of Whitenoise and wrote:
« With the recommended parameters, Whitenoise uses keys with at least 1600 bits randomness. Exhaustive search of 1600 bit keys is completely and absolutely infeasible. Even if we hypothesized the existence of some magic computer that could test a trillion-trillion key trials per second (very unlikely!), and even if we could place a trillion-trillion of these computers somewhere throughout the universe (even more unlikely!), and even if we were to wait a trillion trillion years (not a chance!), then the probability that we would discuss the correct key would be negligible (about ½ to the 1340 power which is unimaginably small). Hence, if keys are chosen appropriately and Whitenoise is implemented correctly, exhaustive key search is not a threat."
« After careful security analysis, I was unable to find any security weaknesses in the Whitenoise stream cipher. Whitenoise resists all of the attack methods I was able to think of. This provides evidence for the security of Whitenoise. »
How do we calculate the length and strength of a Whitenoise key?



The length of a Whitenoise key is calculated by multiplying the length of the subkeys in bytes. Multiplying the 10 smallest prime numbers together to form the smallest Whitenoise key possible would create a key stream greater than 100 billion bytes long. And, we only have to store 158 bytes of key stream information (like DNA) to exactly recreate this key.
The strength of a Whitenoise key is calculated by adding the lengths of the subkeys in bytes and multiplying by 8 bits per byte.

HOW DOES DIVA WORK?

The fundamental characteristic of Dynamic Identity Verification and Authorization and the different functions it serves is the ability to generate and compare tokens ahead in the key stream that have never yet been created or used. These and other similar DIVA techniques are ideal for identity verification, secure network access, continuous dynamic authentication, authorization, signature, inherent intrusion detection and automatic revocation. 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. It is not sending across either an offset or a key with this request.



We are continuously and dynamically comparing tokens to insure 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 token beginning at its last valid offset. Server authenticates the user/device by comparing the received token bit-by-bit to the token generated at the server for this account/person/device beginning at its last current dynamic offset for this key. If they are identical then the Server acknowledges by sending authorization without sending either key or offset information. Both server and endpoint update dynamic offset independently. The system is synchronized for the next continuous authentication query. The account is automatically locked if the comparison of tokens fails.

DIVA encompasses the following abilities: stateful two-way and one-way authentication. Two-way authentication means that each endpoint can request and send authenticating segments of data or offsets. This means that each endpoint has key generation capability. One-way authentication means that only one endpoint (server/site) has key generation capacity. The server then writes back to the endpoint subsequent segments of key stream data that have not yet been used (and delivers this data chunk securely or otherwise)*. On the next session, the server/site compares the actual data at the endpoint to the data they can generate using the endpoint's key structure and current offset. With DIVA, the key stream is polled throughout the session to continually identify and verify that the correct user is on the network. It is possible to incorporate transmission of session keys, use of time stamps, biometrics etc. to increase the security of initial network access (login).

DIVA has stateful detection. The offsets of the key streams must remain in sync between the endpoint and the server. If an interloper manages to steal a key, or gain network access, then the offsets between the server, the legitimate endpoint, and the interloper become out of sync.

There are only two outcomes:

1) The legitimate owner uses his key/card/device first and the segment of random key data (or offset) is updated on the legitimate card. The thief then uses the stolen key /card and it won't process because the data segment (or offset) does not match between the stolen key /credit card and the server. The account is immediately disabled.

2) The thief uses the stolen key/card first successfully. The next time the legitimate key owner tries to access the network they are refused because the stolen card/device has been updated with a new offset or segment of data, the offset on the server database has been updated, but not segment of data or offset on the legitimate card/device. Theft has been identified. The account is immediately disabled. Where the theft occurred is known because of the previous transaction.

DIVA has automatic revocation. The inherent intrusion detection is simply continuing to monitor that offsets and key segments (tokens) always remain in sync. This is a simple comparison of offset numbers or sections of random data. Without any human intervention, the instant out of sync offsets are detected then the account is frozen and that key is denied network access. It does not require going to outside parties, revocation lists etc. A system administrator can remediate or deal with any situation without worry of continued or ongoing malfeasance
DIVA/Whitenoise can perform authorization and DRM. The assignment and monitoring of permissions and usage rights are accomplished by using different portions of the key stream in the same fashion as authentication.

DIVA prevents all known cyber attack classesSummary

• Man-in-the-Middle attacks are prevented because there is no key exchange.
• Side Channel attacks are prevented because all operations are order 1 after key load and because there is no access to the key.
• Mathematical and factoring attacks are prevented because keys are created by a binary mechanical process as opposed to arithmetic ones requiring multiplication and mods.
• Botnet attacks are prevented by configuration with server so the botnet never has access to all the key material to authenticate data being sent OUT of a network or computer.
• Brute force attacks are not feasible with the continually changing dynamic offsets.
• 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.

Future attack capabilities

Quantum computing attacks are prevented because every variable is variable.

---

* Note:
DDKI and DIVA are very flexible while remaining secure and allow for context and goal specific configuration. The anti-botnet configuration above is an example.
No private key – Banks and governments in particular might want a system design where they never give the private key to their client. Offsets and tokens are opposite sides of the same coin. The offset is the pointer into the exponential key stream for the token; and vice versa. In a one-way authentication configuration, the endpoint will have the next token on their device or card for secure network access and authentication but they do not have either key or offset material present on the device. That, along with additional multifactor authentication, then renders any possible theft irrelevant. And the client cannot give their key away.
**

A tunnel paradigm is another configuration that allows distributed keys to securely distribute more distributed keys to enroll new clients in real time. This overcomes the encumbrance of having to physically copy a distributed key to each new endpoint in a secure network and makes scalability dynamic and simple.
Additionally, since it can be deployed at the data link layer it overcomes the encumbrance of having to configure the security for every use at the application layer. This, along with being bit independent and data agnostic also makes these systems interoperable.

Information

Euratechnologies

CEIS