White Paper: The Internet of Things and Whitenoise Technologies

White Paper: The Internet of Things and Whitenoise Technologies

Author André Brisson 2014 www.wnlabs.com

About the Author: http://www.wnlabs.com/about/executive.php

What is The Internet of Things?

"The **Internet of Things** (**IoT**, also **Cloud of Things** or **CoT**) refers to the interconnection of uniquely identifiable embedded computing-like devices within the existing Internet infrastructure. Typically, IoT is expected to offer advanced connectivity of devices, systems, and services that goes beyond machine-to-machine communications (M2M) and covers a variety of protocols, domains, and applications.[1] The interconnection of these embedded devices (including smart objects), is expected to usher in automation in nearly all fields, while also enabling advanced applications like a Smart Grid.[2] "From Internet of Things Wikipedia

The hope and the necessity of the Internet of Things is that it will make societies, enterprises and all of us more efficient particularly in our consumption of critical resources. While it is nice to have a refrigerator that will tell you what to pick up at the store ultimately this is not a critical benefit, particularly when that capacity could be opening vulnerabilities into more important systems.

Deployed correctly, and securely, the Internet of Things can help us regulate and become efficient in the use of critical services like transportation and critical resources like energy. (For example, it doesn't make much sense to dump excess electricity generated when it can be redirected to locations that don't have enough of it.)

In reality, the Internet of Things has already been in construction and little adequate thought and planning has been given to security and the frameworks, protocols and network design that is governing them all. Information Technology is rushing forward ignoring "unintended consequences."

Thankfully "prominent standardization bodies, such as the IETF , IPSO Alliance and ETSI, are working on developing protocols, systems, architectures and frameworks to enable the IoT." From Internet of Things Wikipedia

Characteristics and challenges of securing components in the IoT

The majority of components comprising the Internet of Things are highly sensitive to costs of manufacturing. These devices are generally low cost components with little access to power and limited memory and/or storage.

These dramatic physical limitations of devices are further complicated by haphazard connectivity. We tend to think in terms of a "client-server" context and assume that all devices like cell phones can communicate directly with a server. In this way of thinking we want to believe that the network server and the administrator running it can exercise some reasonable level of control in its security. In the Internet of Things there are a plethora of components that have limited communication with other components but not directly to a server. For example there are products where there are multiple chips on a board that can communicate with one another but not directly out to a server. This opens a huge area of vulnerability where one component can be compromised and in turn compromise other components without visibility to a network or network administrator.

How are they trying to secure the Internet of Things?

There appear to be two primary ways that are being attempted to secure the Internet of Things. Both are flawed.

1. Embedded security is the attempt to put meaningful security directly into tamper resistant chips and microprocessors. Current security is obviously vulnerable as illustrated through the rampant theft and breaches to which we have grown accustomed. Frankly, affordability of devices and the nature of business competitive advantage mean that no one is going to put a $10 or $20 chip set into a $10 device. Even if they did it would be ineffective because of the physical limitations of power, processing and memory. It is barely possible, if at all possible, to run 128 bit encryption or asymmetric key security controls in this environment.

The explosion of the number of components and devices in the IoT seems to follow Moore's Law. Complicating matters is the explosion of media and the need and demand for faster and faster processing and streaming speeds.

With current cryptography (non-Whitenoise) there is an inverse relationship between speed and security. We saw this in the performance of Twofish and Blowfish. In making Blowfish more secure Twofish took a performance hit.

We see this happening again today where major chip companies are beginning to adopt AES NI (New Instructions). The reason is to improve the speed of applications using the Advanced Encryption Standard (AES).

There is a lot or research doing performance analysis on the increased speeds (more speed, more speed, more speed) and a dearth of research on the security implications. It appears that concessions are being made for speed of processing and the cryptographic community is turning a blind eye to analyzing the security implications.

And even with this concession to learned helplessness in IT security the AES NI are not even that fast and will again have trouble keeping up without accelerators. One performance analysis measured improvement of throughput as going from 28 clock cycles per byte to 3.5 clock cycles per byte. Compare that to the performance of Whitenoise in FPGAs that can process 2 bytes per clock cycle. Comparatively AES NI would need 14 clock cycles to process 2 bytes in the same time that it takes Whitenoise to process the two bytes in one clock cycle. And in simple old generation Xilinx FPGAs Whitenoise has encryption speeds that can easily reach 1.28 Gb/sec and with little more cost reach 12.8 Gb/sec without any acceleration. www.wnlabs.com/pdf/Whitenoise_in_FPGA.pdf

As the big chip manufacturers are moving to AES NI they leave themselves vulnerable to competitive advantage. The demand for simple, inexpensive chips for the IoT can easily create a scenario we have seen before but with much bigger consequences. At one point Texas Instruments overtook Intel as the largest seller of chips globally when Nokia adopted a simpler, less expensive chipset for parts of their product line.

2. Data encryption at the transport layer with SSL and TSL is the complimentary component. Asymmetric network security has been proven to be ineffective at the application layer. Hiding it down at the transport layer is akin to believing that the “bad guys” cannot find the deep web. Public key systems have always been vulnerable to a host of attack classes including Man-in-the-Middle attacks. This is exasperated in M2M and IoT environments where there is not even a person that might sense something is amiss.

Deploying security at a lower level of the Internet is a double edged sword. On the upside it means that there is less implementation of security one-at-a-time at the application layer. Everything above the security (SSL and TSL at the transport layer) is “protected”. However, we are all aware of their vulnerability as we suffer unprecedented levels of theft and cyber crime. Therefore, everything above poor encryption and identity management at transportation layer is vulnerable and increases the number of targets exponentially in the IoT. Security flaws at this level compromises the security of everything above it.

Just this week news reports indicate that a Russian gang has been able to steal 1.2 billion user names and passwords from Internet sites giving access to additional personal information. This is one of the areas that SSL and TSL are supposed to protect. This breach represents up to one sixth of the world’s population being hacked.

We will examine another option of distributed keys in turn distributing more distributed keys later in this paper.

What are the concerns?

As in all current networks, viruses and malware are fairly easy to introduce into M2M components where there is little to no communication with a server for oversight and little or no human oversight.

It is difficult to quantify the exact degree of vulnerability we are facing as societies and individuals but many feel that we are at the brink of digital Armageddon. For example, the US power grid is essentially constructed in just a few interconnected grids. It is a challenge to prevent the cascading effects of damage. We saw that in 2003 when the North Eastern United States blacked out because of a “branch hitting a power line” or a “lightning strike”.

A sad human analogy of how viruses can cascade into a tragic situation is the current efforts of world health organizations and governments in preventing the spread of the Ebola virus.

What is the proof that the Internet of Things is already here and we are already vulnerable?

Countries are already engaged in cyber warfare. We will only list a few selected examples although these kinds of breaches occur daily. Criminals are also actively involved in cyber crime and it has been estimated that these behaviors cost the global economy almost half a trillion dollars in losses in 2013.

- Idaho Labs generator ruined by a virus
- Russia attacking Estonia
- Suxtnet and Iranian Nuclear ambitions

White Paper: The Internet of Things and Whitenoise Technologies

What are the minimum requirements to secure the Internet of Things?

- Every device or component on the IoT needs to be uniquely identifiable with a key that cannot be broken or spoofed. [Solution: Whitenoise keys]
- Every device or component must have a virtually manufactured protocol to keep cost of security to an absolute minimum. [Solution: Dynamic Identity Verification and Authentication (DIVA)]
- Every device or component must be able to join a virtual framework. [Solution: Dynamic Distributed Key Infrastructures]
- The protocol must be able to be electronically provisioned to include legacy components and devices to bring them into secure networks.
- The solution must be easily scalable. (Solution: DDKI)
- The solution must be interoperable across all operating systems and topologies [Solution: DDKI and DIVA]
- The solution must be self monitoring (intrusion detection) [Solution: DIVA]
- The solution must be self healing (automatic revocation and updating). The first US National Cyber Leap Year Summit (NCLYS) described this as a “biological defense” akin to a person’s immune system
- The solution must not require the removal or replacement of any other devices or security controls
- The solution must operate with any other security controls at different parts of the network
- The solution must operate with near zero overhead and processing
- The solution must operate with near zero latency
- The solution must operate at maximum possible speed for any component
- The solution must be dynamic. The first US National Cyber Leap Year Summit described this as a **moving target defense**
- Embedded security (chips/microprocessors) must have top level security even in chips that are simple and cheap
- Embedded security must work with firmware (upgradable)
- Embedded security must have ability to write back last valid token (key) or offset for DIVA
- Every device must have some level of communication ability
- Every device or chip must have some level of memory or storage capacity

Whitenoise Technologies address the current short comings of security in the IoT

For the balance of this paper we are going to use a security analysis of the Internet of Things conducted by HP. Subsequent articles about this study have said that the realities it highlights can severely hamper the safe growth of this massive, dynamic vertical market.

- HP security study finds the average device has 25 security flaws

"The researchers focused on some of the popular devices from the IoT categories. These include garage door openers, scales, home alarms, door locks, hubs for controlling multiple devices, sprinkler controllers, remote power outlets, home thermostats, webcams, and TVs." From the above HP study.

This paper refers to the Top 10 Security Issues for the Internet of Things as defined by The Open Web Application Security Project (OWASP). We will examine the Top 10 security flaws for the IoT and provide an explanation of how Whitenoise Technologies (Whitenoise keys, DIVA and DDKI) mitigate against or prevent these security holes. https://www.owasp.org/index.php/Top_10_2014-I1_Insecure_Web_Interface

- I1 Insecure Web Interface
  - Weak credentials
  - Weak passwords
  - Capture plaintext credentials
  - Internal and external vulnerability
- I2 Insufficient Authentication/Authorization
  - Weak passwords
  - Weak password recovery
  - Poorly protected credentials
  - Internal and external vulnerability
- I3 Insecure Network Services
  - Attack vulnerable network services
  - Attack device itself
  - Bounce attacks off of the device
  - Buffer overflow attacks for Denial of Service
  - Sniffers and fuzzers
- I4 Lack of Transport Encryption
  - Easy view of unencrypted data passing between or over networks
  - Traditional crypto vulnerabilities associated with SSL and TSL i.e. Man-in-the-Middle attacks etc.
  - Compromised Transport Layer means everything above it is vulnerable
- I5 Privacy Concerns
  - Insufficient authentication
  - Lack of transport encryption and storage of data in encrypted format
  - Insecure network services
  - Collection of unnecessary personal data

- I6 Insecure Cloud Interface
    - Insufficient authentication
    - Lack of transport encryption and storage of data in encrypted format
    - Attack likely from the Internet
    - Easy to guess credentials
    - Using password reset mechanism to see if account exists
    - Identify is SSL is in use
    - Account enumeration
- I7 Insecure Mobile Interface
    - Insufficient authentication
    - Lack of transport encryption and storage of data in encrypted format
    - Attack likely from the Internet
    - Easy to guess credentials
    - Using password reset mechanism to see if account exists
    - Identify is SSL is in use
    - Account enumeration
- I8 Insufficient Security Configurability
    - Lack of granular ability to configure authorizations
    - Weak passwords and credentials
- I9 Insecure Software/Firmware
    - Insecure firmware software unencrypted updates
    - DNS hijacking
    - Malicious updating
- I10 Poor Physical Security
    - USB, SD cards, other storage devices that give access to the Operating System

Summary of the security flaws HP found in the IoT

A review of the most glaring ten security flaws in the IoT predominantly center around poor authentication and weak credentials (identity of devices and persons), poor authorizations (i.e. LDAP and CAS) and network services, and either no use or poor implementations of SSL and TSL at the transport layer. It should be noted that even with proper deployment of SSL and TSL that they are still vulnerable to several attack classes. We will examine a much better distributed alternative to SSL and TSL in a moment.

The fact that security is being compromised by poor choices and the continued predominant reliance on the use of passwords as the primary security control; and the use of poor and weak passwords in general to compound that state is really nothing short of unbelievable in 2014 as well as unnecessary.

Weak or improperly used credentials like certificates are also listed as a cause of problems. Whitenoise provides perfect certificateless authentication because a person or device is identified by a unique key and by their unique encryption.

You will find a discussion of how network services like LDAP, CAS and IPv6 can easily be secured and become an effective method of ubiquitous secure key distribution.

How Whitenoise Technologies Address security flaws in IoT

Historical reasons of how we got to the state we are in

To understand how Whitenoise Technologies can easily address all the short comings of security in the IoT we must first understand how we got here.

The world has always operated with distributed systems, distributed identity and distributed security. You are given your driver's license. A soldier is given his gun.

With the advent of computers and soon connected computers (networks) we immediately ran into the problem and historical stoppers of distributed security and identity. These stoppers were key management, key distribution and key storage.

Secure distributed networks could not manage the exploding key number problem. Historically, the number of keys needing to be stored in a secure distributed network was the square of the number of end points or nodes. This meant that a secure network with 10 endpoints required storing 100 keys. Whitenoise technologies solved this problem because there is a one to one relationship between the number of endpoints and the number of keys to store.

The second historical stopper of secure distributed networks was the storage volume of keys themselves and the key exchange of these keys. Big keys are obviously better than small ones for identification, authentication, and encryption. And if one wants to retain the characteristics required for a one-time-pad these keys needed to be random and larger than the data to be encrypted. The technology did not exist back then so the keys were small and repeated themselves when used to encrypt data which by definition makes them breakable.

Also with slow networks exchanging keys even insecurely was a laborious process.

Both of those historical stoppers have been solved because Whitenoise and DIVA is a protocol that creates an infinite number of one time pads WITHOUT the exchange of key material (or offsets) after the initial one time key distribution of a single key. This was solved by the exponentialism of Whitenoise keys. See Addendum.

So, a brilliant (though limited) solution to these problems was created in Public Key asymmetric systems. These systems however still had fatal flaws like vulnerability to Man-in-the-Middle attacks, ability to be factored with sieve or brute force attacks, as well as mathematical attacks and cryptanalysis. For the time, however, it was sufficient because processing speeds were so slow it in effect made many of these flaws infeasible to attack in any meaningful or useful timeframe.

Now with incredible processing speeds these Achilles' heels make us vulnerable even to children let alone nation states. Note the recent disclosure that a Russian gang has been able to steal at least 1.2 billion user names and passwords supposedly sufficiently protected under these protocols. That is 1/6 of the world's population that has been compromised.

Conceptually, imposing an asymmetric security system upon distributed networks seems at odds with simple network design. Our networks are distributed systems. Just picture Google, Facebook and Amazon adding more and more servers as they increase the range of their distributed services.

Similarly, Dynamic Distributed Key Infrastructures (DDKI) is a scalable, interoperable, tiered, hierarchical, virtual, distributed network framework made up of devices and components that deploy Whitenoise keys and the DIVA protocol.

What are the characteristics of Whitenoise?

executive summary

One distributed key will create an infinite number of one-time-pads (OTP) satisfying the three characteristics that make OTP the only mathematically provable unbreakable key technology:

- The key is random. See UVIC performance analysis where there were not even anticipated statistical errors in testing for randomness and entropy against the NIST test suite.
- The key is larger than the data to be encrypted or protected. See exponentialism in the Addendum below and make a key yourself with the downloadable utility.
- And the key is only used once. Review how DIVA operates in the addendum.

Whitenoise is unbreakable. David Wagner, a noted cryptography performed a security analysis and could find no mathematical attacks and concluded about the only attack left was brute force:

"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 $\frac{1}{2}^{1340}$ 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."

David Wagner University of California, Berkeley and expert witness on cryptography for the US Congress.

A Whitenoise key has never been broken by any government, university, hacker and security groups like Black Hat and DEFCON, or any individuals. See the $200,000 contest that is just concluding.

It is also critical to note two things:

Whitenoise technologies are not a zero sum game. They work seamlessly with public key asymmetric systems. When used together it creates a two channel (both symmetric and asymmetric) challenge where one of the keys is dynamic. A hacker would have to break two keys simultaneously for each and every breach attempt. And, one of the keys is changing rapidly. See the Black Hat contest clock. This means that anyone that is mandated to use AES can and be fulfilling their mandate. Whitenoise technologies are then added without impediment to fix the flaws attendant with the mandated technologies.

A single Whitenoise key provides for ALL key based security controls including:

- Perfect identity
- Secure network access
- Continuous dynamic authentication
- Authorization
- Signature
- Non-repudiation
- Inherent intrusion detection
- Automatic revocation
- Encryption

YOU DO NOT NEED TO REMOVE OR REPLACE ANY OF

YOUR EXISTING SECURITY CONTROLS OR FRAMEWORKS.

Why is Whitenoise resistant to all attack classes?

First, Whitenoise can replace having to use passwords at all with externally deployed keys or biometrics. In contexts where there is no access to the device the key itself is sufficient.

Each key is unique to each device, endpoint or person. It cannot be spoofed and the keys are of such extraordinary size and associated with specific devices that they can't be given away nor would they be useful if it was possible to steal a key. If one wants to continue using passwords for a multi factor authentication routine any compromise of a password remotely has no effect on the Whitenoise identity management key because they are completely unrelated. Please study the addendum for a look at the use of biometrics and Whitenoise to remove reliance on passwords.

Dynamic Distributed Key Infrastructures (DDKI) is a virtual framework that creates a secure network of networks of devices, endpoints, and servers that deploy DIVA. It is completely interoperable and forever scalable (both the keys themselves and the virtual network.)

This single protocol and single framework eliminates all known attack classes:

• Man-in-the-Middle attacks are prevented because there is no key or offset exchange

• Side Channel attacks are prevented because all operations are order 1 after key load and because there is no access to the key

• Botnet attacks are prevented by configuration with second server so the botnet never has access to the entire key and offset information.

• Quantum computing attacks are impossible because every variable is variable.

• Brute force attacks are prevented because the keys can't be factored.

• Denial of service attacks can be prevented by exploiting unbreakable identity and secure network access so that hackers could never get on a network and flood it.

Please review and study the DIVA protocol in the Addendum.

How Whitenoise Technologies Address flaws in embedded security

After key load every operation in Whitenoise for encryption is an order 1 process, XOr. This is the fastest function available on a computer which means that Whitenoise processing speeds are limited only by the physical characteristics of a chip or computer or device and not by any inherent characteristic of the protocol. Whitenoise will always stay ahead of the threat curve.

It operates with no measurable latency and the absolute minimum amount of processing possible. It is also bit independent which means there is never any reframing or restarting if bits are flipped. And it means that keys and streamed data can be parsed and reassembled.

Note that asymmetric and AES processes require complex mathematical or processing routines that chew up available resources and make their overhead costs explode. Whitenoise is a mechanical process predominantly using XOr which by definition makes it the fastest and simplest option available.

It is possible to deploy Whitenoise for DIVA with exponentially stronger and faster keys with a minimal footprint.

It is the only national security level strength key technology that could be deployed in chips not much more sophisticated than those found commonly in children's toys.

Please review: www.wnlabs.com/pdf/Whitenoise_in_FPGA.pdf

The only requirements for Whitenoise and DIVA are that there is connectivity, and write back capacity to track current dynamic offsets to firmware working in conjunction with the chip.

How Whitenoise Technologies address flaws in the approach of transport layer encryption

To understand the preferred option of DDKI, DIVA and Whitenoise for encryption of data in transport of data at the data link layer please study the publicly available practicum that won the Cool Science award for all university projects by BCNET.

www.wnlabs.com/Tunnel_Distributed_Keys_distributing_more_keys.pdf

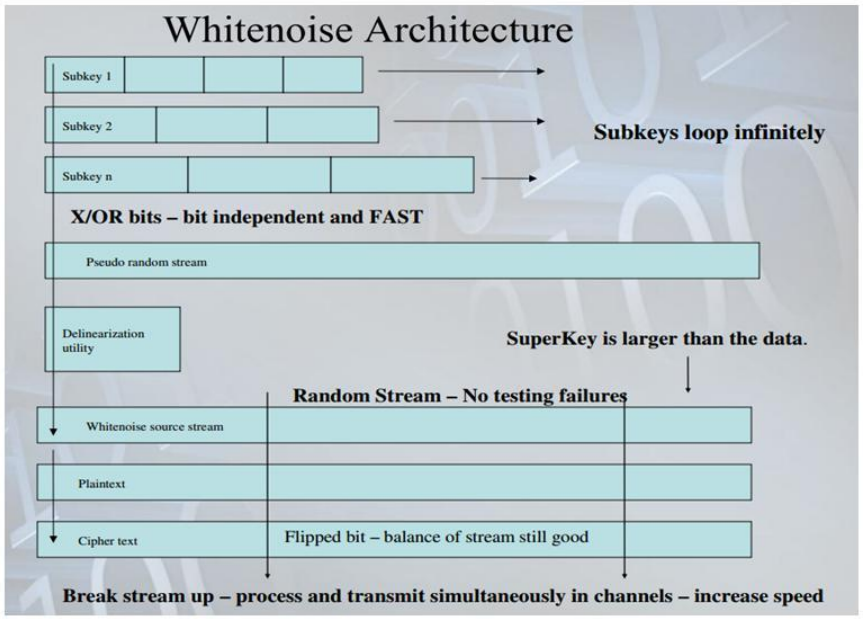
Note: You can continue to use SSL and DSL at the transport layer for an addition factor in security without replacement, removal, or direct integration. Note: This is patented technology. The patents are publically available as well.

Addendum

How is a Whitenoise key made?

[redacted]

How is a Whitenoise key made?



- variable number of prime number length subkeys
- each bit is XOr'd with the corresponding bit of the next subkey
- it is run through an S-box to create a one-way function
- it becomes first byte of delinearized key stream

NOKIA

[redacted]

The subkey lengths themselves are populated with random data so that they do not operate like line feed shift registers or counters.

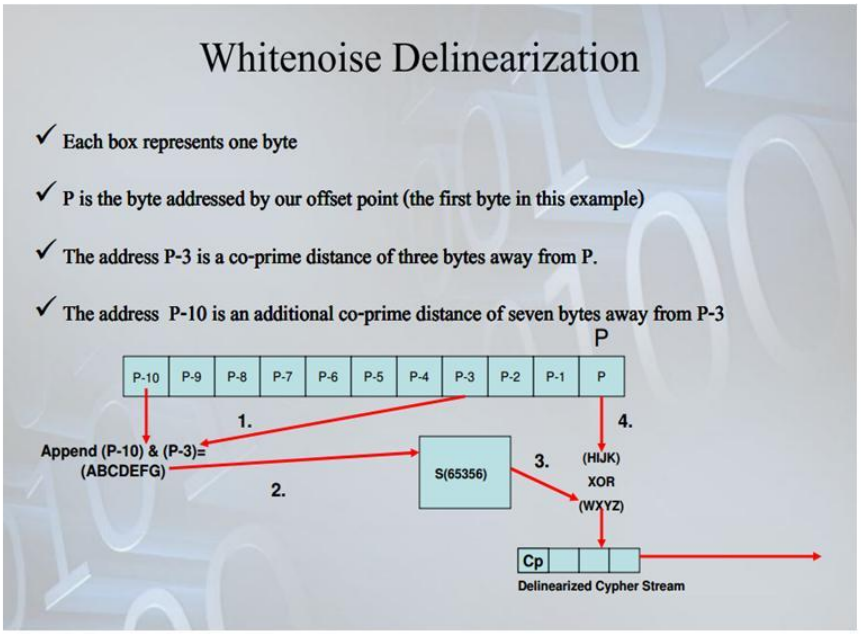
Learn how Whitenoise keys are made: https://www.youtube.com/watch?v=51gqPzEHXSA

See a demo of how WN keys are made and speed testing.

https://www.youtube.com/watch?v=9Ebgya6lxS4

What are the one way functions that make Whitenoise unbreakable?

What are the Whitenoise one-way functions?



- two bytes are taken from the initial key stream, appended together and pushed through an S-Box
- only one byte emerges
- a hacker cannot go backwards and guess two bytes from one byte of information
- the hacker has no knowledge of the number of subkeys

NOKIA

The hacker has no knowledge of the number of subkey used to create the initial key stream. Neither does he have any knowledge of the random data that is used to populate these prime number subkey lengths.

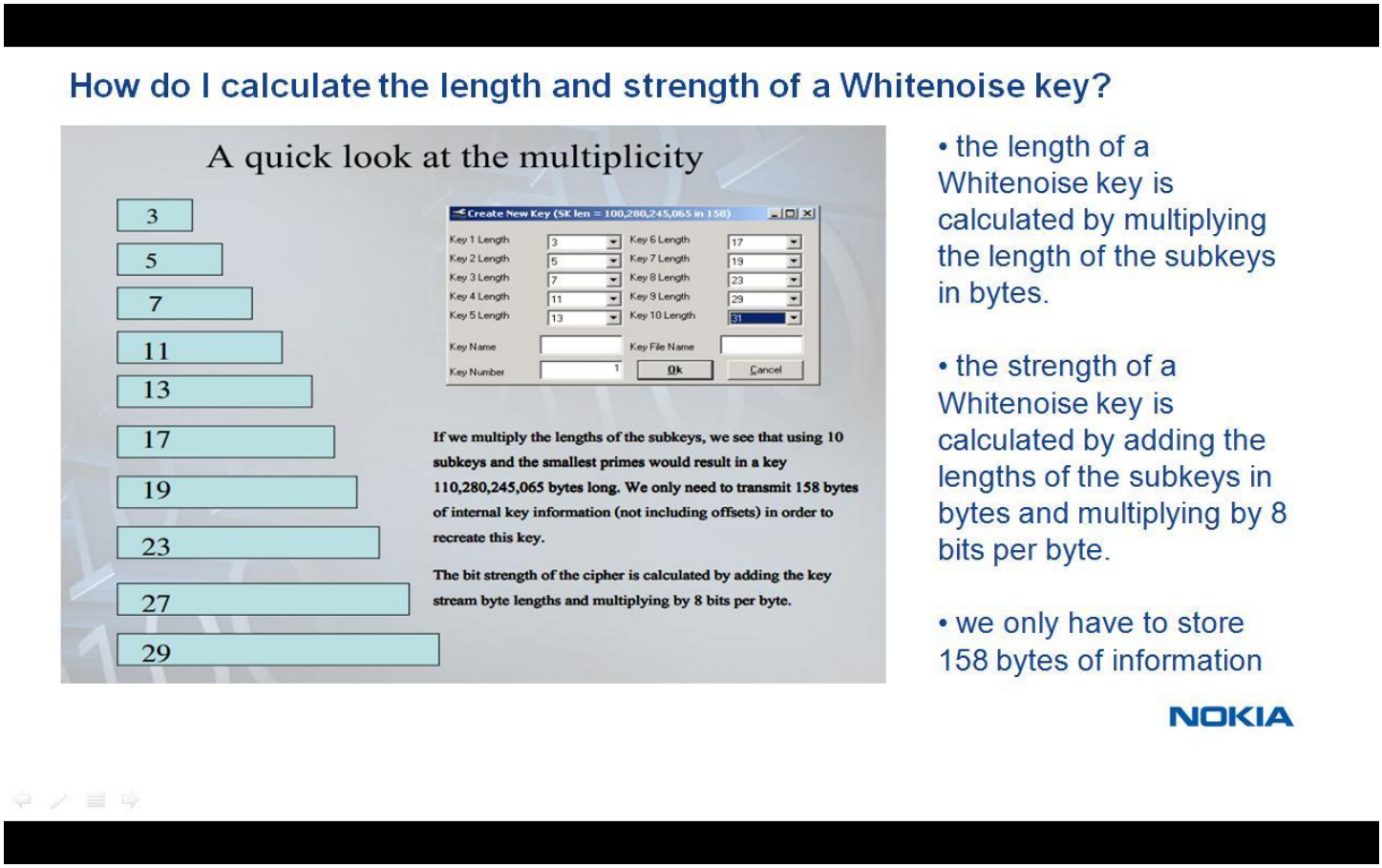
Note:

- if the algorithm operated only horizontally it would be flawed and acting like a Line Feed Shift Register.
- If the subkey lengths operated with data in an ascending fashion i.e. 1,2,3,4,5 it would be acting like a counter and would be flawed.

The most obvious reason that it is unbreakable is that it is a one-time pad where there is no transfer of key or offset information. The three requirements for a one time pad are:

1. The key is random
2. The key is larger than the data to be encrypted or protected
3. The key is used only once (it's dynamic)

How do I calculate the length and strength of a Whitenoise key?



The example above created a key stream over 100 billion bytes long and only 158 bytes of key structure information needs to be stored to recreate it deterministically.

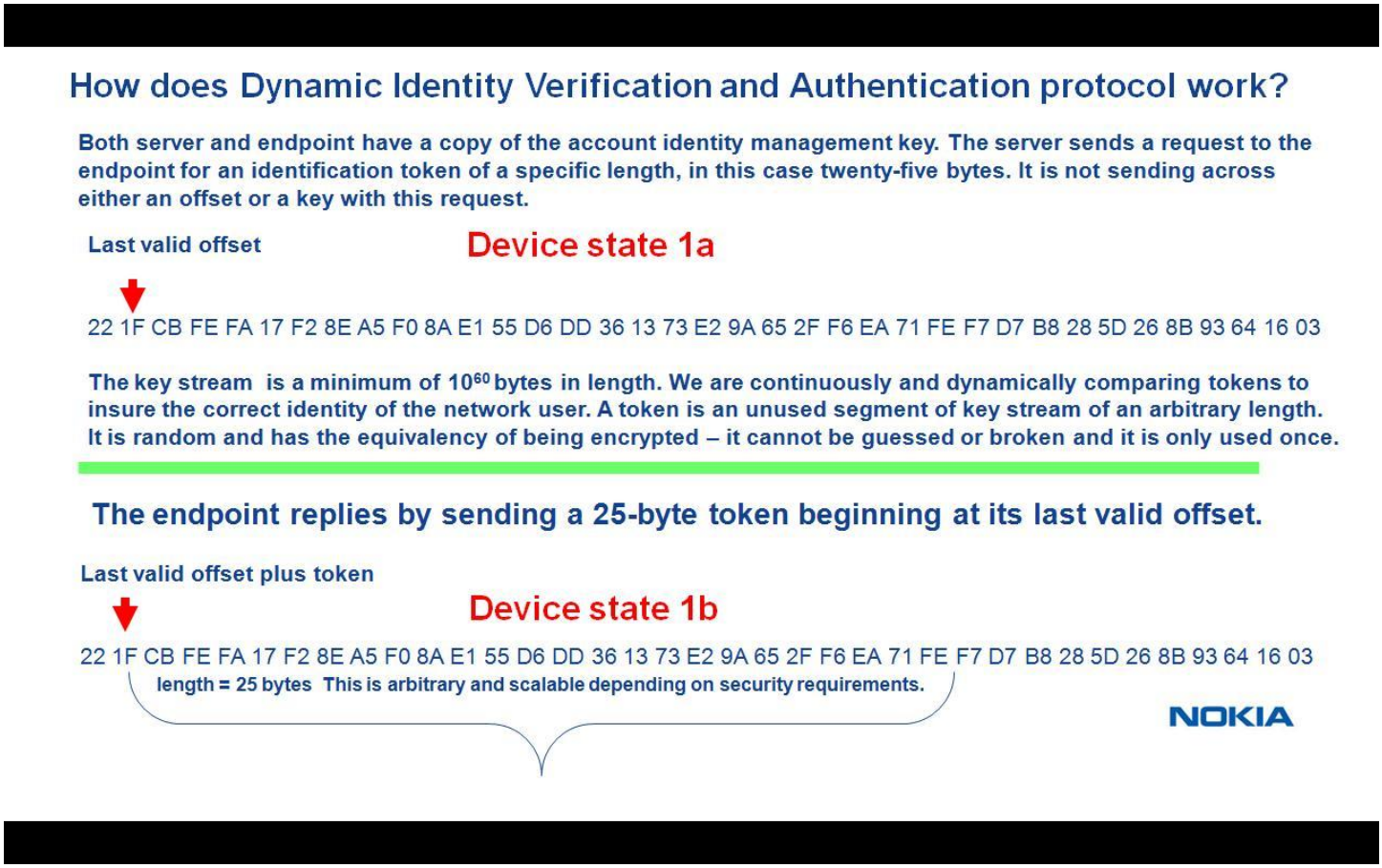
Make and test a key yourself.

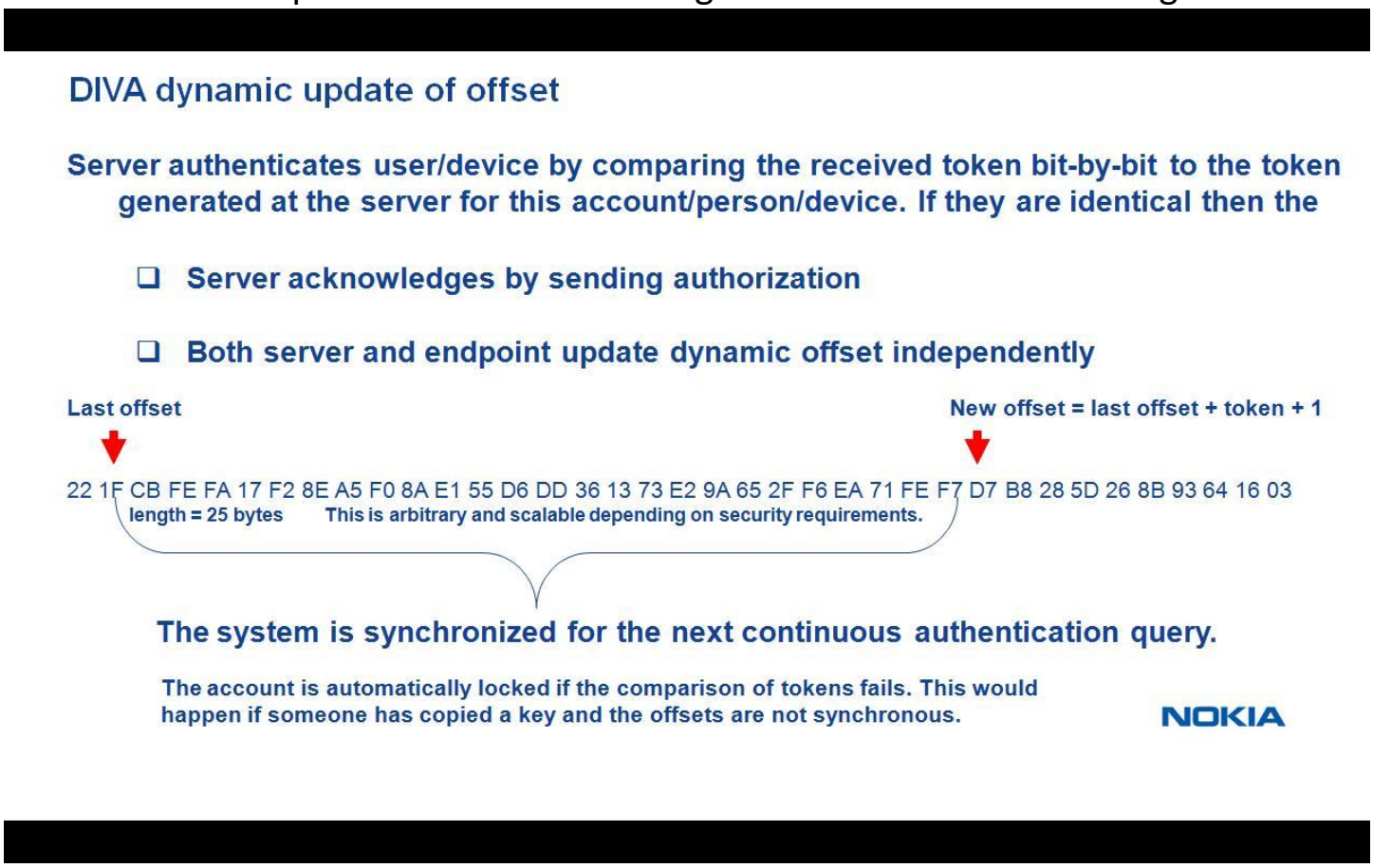
Test before you write!

YouTube demonstration - https://www.youtube.com/watch?v=9Ebgya6lxS4

Download - http://www.wnlabs.com/technology/Self_Demo.php

How does DIVA work?





Learn how DIVA works tutorial: https://www.youtube.com/watch?v=c6qaKkV9GJU

See a demo of how DIVA works: https://youtu.be/czRC2Js3lG8

See a demo of how DIVA works in Sierra Wireless Best in Class:

http://www.wnlabs.com/technology/sierraDemo.php

Replacing passwords with biometrics combined with Whitenoise Identity Management keys

This paper has discussed the effects of cascading damage to breaches in the power grid. One goal of any adequate security framework for the IoT or CoT should effectively prevent the cascading effect.

Additionally, the **Basket of Remotes** concept where we will have "hundreds of applications to interface with hundreds of devices that don't share protocols for speaking with one another" was examined. We can look at an individual's or enterprise's group of devices in the IoT as a distinct group, or what we will now call your personal **Community of Things** which is a subset of the IoT. And we saw from the HP study these can include things like garage door openers, scales, home alarms, door locks, hubs for controlling multiple devices, sprinkler controllers, remote power outlets, home thermostats, webcams, and TVs."

While most of these products are very cheap and inexpensive and highly sensitive to cost of manufacturing and with very limited power, processing and memory/storage, not all of them are. Virtually everyone has a TV or a computer or a tablet or a phone with enough resources to easily process biometric authentication.

The fear of the use of biometrics is that if your physical identity is stolen then your identity would be compromised for your entire life. This need not be the case as was discussed in a Gartner Vendor Briefing of turning a biometric into a one-time-pad with Whitenoise keys.

Whitenoise keys are highly unique with one characteristic being that it is quite easy to create and store key structure that will generate deterministic random key streams that are easily quadrillion of bytes long. This exponentialism characteristic is discussed in papers written for the first US National Cyber Leap Year Summit: Vision and Game Changer. This was also discussed in a paper for the United Nation's Organization for Economic Cooperation and Development.

Because the keys are so long, different static portions of the same unique key (specific tokens) are used to identify a person, their devices, and the services or accounts they use i.e. their iPad, their phone, their notebook, their desktop, health insurance number, driver's license, passports, banking accounts, unique commercial services etc. The balance of the key is used to create a dynamic one-time-pad.

This can easily address the idea termed **Basket of Remotes** and create what we will now call a **Community of Things** (your digital stuff.) One of a person's devices will have enough capacity to run a biometric authentication and pass a positive authentication call to other things that have limited power, memory/storage, and processing capacity. These limited capacity devices will have the minimum ability of receiving a Pass/No Pass statement without having to process the actual biometric authentication routine which is performed by another connected device. With a positive authentication pass, the device's Whitenoise key is then invoked as the second authentication factor that then continues throughout its use or session to provide dynamic security and identity for that device that is part of their **Community of Things**, which in turn is a subset of the Internet of Things or Internet of the Cloud.

This approach can create logical subsets or groups of devices (Community of Things) and any potential breach then limits damage or comprise to a much smaller and manageable set/group of devices.

Two further benefits are the binding of organic identity (everyone is unique) with a unique digital key using Level 3 or 4 Identity Proofing and removes the necessity of using either removable keys on USBs, SKs etc. which can easily be lost. Because external keys can easily be lost or forgotten, people have a tendency to leave them in the device which defeats the purpose.

In either event, the use of external keys or biometrics in conjunction with Whitenoise removes the need to rely on passwords which are generally not used properly and easy to compromise.

IPv6 as a key distribution mechanism with unique addressing and size considerations

Jean-Louis Gassée predicts that the most likely problem we will face moving forward is what he calls the **"Basket of remotes"** problem, where we'll have hundreds of applications to interface with hundreds of devices that don't share protocols for speaking with one another. https://www.owasp.org/index.php/Top_10_2014-I1_Insecure_Web_Interface and Wikipedia.

DIVA and DDKI can be this unifying protocol and framework. And, possibly the easiest and most thorough way of distributing keys (and yes in a uniquely encrypted state) could be with the uptake of IPv6 addressing.

IPv4 only created **4.3 billion** IP addresses, an example of poor foresight. These unique addresses were quickly exhausted. When we consider that there are about 7 billion people on earth and hundreds upon hundreds of billions (if not trillions) of connected devices smart phones, tablets, computers and all the components of IoT you can see there must be a lot of repetition in addressing.

The new protocol slowly being vended in, IPv6, can assign about 340 trillion trillion trillion unique addresses. That is a good start. However, addressing does not remove any of the current network vulnerability we suffer from. The networks deploying IPv6 is still vulnerable to the same security demons.

Addressing does not address the problems attendant with embedded security.

Addressing does not address the problems attendant with flawed transport encryption.

Addressing does not eliminate any of the traditional network security flaws. Without belaboring the point, the following papers describe vulnerabilities with IPv6.

Addressing could however be a good key distribution mechanism.

--

http://www.sans.org/reading-room/whitepapers/detection/complete-guide-ipv6-attack-defense-33904

The above paper describes the following problems with IPv6.

- Address enumeration
- Port scanning
- Stack based overflow
- Protocol vulnerability

- MiM
- Sniffing
- DoS
- Smurf attack
- Duplicate address detection
- TPv6 fragmentation

http://www.darkreading.com/attacks-breaches/five-security-flaws-in-ipv6/d/d-id/1128758?

The above paper lists the additional vulnerabilities.

1. Trespassing
2. Filtering device bypass
3. Denial-of-service (DOS)
4. Anycast: Not safe anymore
5. IPv6 puts IPv4 at risk

From Wikipedia

"As we move (or have moved) to **Internet Protocol Version 6 address** (**IPv6 address**) which is a numerical label that is used to identify a network interface of a computer or other network node participating in an IPv6 computer network. An IP address serves the purpose of uniquely identifying an individual network interface of a host, locating it on the network, and thus permitting the routing of IP packets between hosts. For routing, IP addresses are present in fields of the packet header where they indicate source and destination of the packet.

IPv6 is the successor to the Internet's first addressing infrastructure, Internet Protocol version 4 (IPv4). In contrast to IPv4, which defined an IP address as a 32-bit value, IPv6 addresses have a size of 128 bits. Therefore, IPv6 has a vastly enlarged address space compared to IPv4"

Consider – the weakest key strength that Whitenoise deploys is > 250,000 bits and generates unique, deterministic, random key streams greater than ten to the sixtieth power (10^{60}).

If we were to deploy Whitenoise in conjunction with an IPv6 address to create a one-time-pad IP address then likely no other version of this protocol would be needed until the end of time because of the existing security concerns. And, if our technologies are deployed from this level and vantage point, we could effectively identify and secure everything that uses it and these endpoints, clients, applications, systems, etc. wouldn't even know

that the security layer was there.

Similarly other network services like LDAP and CAS can deploy Whitenoise technologies and rapidly secure a larger base.

AES NI from Wikipedia

Intel AES-NI is a new instruction set for chips that accelerates AES encryption / decryption by implementing computationally intensive steps of the AES algorithm into the chip as opposed to the firmware or software. In addition it is claimed that it improves the security of AES encryption / decryption by addressing the risk of side channel attacks associated with traditional software implementations since the instructions run from the chip which is assumed to be inaccessible.

In doing this they are at least not using lookup tables which are part of the approved AES algorithm.

Even in this posting from the International Archive of Cryptographic research this claim is made as an aside without any test data supporting it.

http://eprint.iacr.org/2010/576.pdf

Since in fact this is a change to the originally approved algorithm, after many years one would expect to find study after study validating this new approach.

Typing “Security Analysis of AES NI” into Google generates many studies about performance.

But why haven’t there been dozens of independent security analyses about this algorithm in the last seven years?