Dynamic Distributed Key Infrastructures and Dynamic Identity Verification and Authentication(DIVA)

(DIVA) and Turning Biometrics into a One-Time-Pad

FIC Hot Topics One-by-One

André Jacques Brisson Whitenoise Laboratories Canada Inc. WNL Vancouver, BC/Cambridge Innovation Center, MA.
abrisson@wnlabs.com

André Jacques Brisson
Whitenoise Laboratories Canada Inc. WNL
Vancouver, BC/Cambridge Innovation Center, MA.
abrisson@wnlabs.com

I. INTRODUCTION FIC 2015

The approach to this paper will be somewhat unique since The International Cybersecurity Forum (FIC) "forms part of a thinking and exchange process that aims at bringing together the full array of stakeholders and decision-makers of the cybersecurity sector."

These stakeholders are politicians and policy makers, military/government and law enforcement officials, the largest commercial entities that are involved in cyber (and all the other commercial entities that need decent working security), academia, students and citizens.

Technology is rapidly changing our world and constantly raises issues that are fundamental to governance, morality, and ethics: i.e. Wiki Leaks, Snowden, terror radicalization videos, etc.

Some cyber technologies raise questions for constitutional lawyers, politicians and policy makers in terms of freedom of speech, structure of government and the critical question of our age: How do we balance Privacy and Security?

This paper will show you some ways to balance privacy and security.

We will first look at Dynamic Distributed Key Infrastructures (DDKI) a secure, virtual, scalable framework. We will look at one technique where distributed keys can in turn distribute more distributed keys and establish these secure links with persons or devices yet to receive a key.

Next, we will examine Dynamic Identity Verification and Authentication (DIVA), a virtual protocol that prevents all known cyber attacks and performs all network security controls. As an example we will look at how DIVA and DDKI can be used to turn biometrics into a one-time-pad and to create secure, adaptive mobile networks.

Finally, we will then look at the agenda for the International Cybersecurity Forum (FIC 2015) question-by-question and comment on how Whitenoise technologies impact each topic area.

II. DYNAMIC DISTRIBUTED KEY INFRASTRUCTURES

Abstract- 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.

Dynamic distributed key infrastructures and dynamic identity verification and authentication address all the articulated needs securely, simply and inexpensively. 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.

III. DYNAMIC DISTRIBUTED KEY INFRASTRUCTURES AND A SCALABLE VIRTUAL SECURE NETWORK FRAMEWORK

Dynamic Distributed Key Infrastructures (DDKI) are virtual networks comprised of components using Dynamic Identity Verification and Authentication (DIVA). DIVA Dynamic Identity Verification and Authentication impede all cyber security and identity theft attacks by demanding proof of identity at the time of network access and throughout the network session.

A network user is provided a unique, unbreakable, one-time-pad identity. The authentication calls occur at a rate faster than is possible for a hacker to breach the network. The identity is constantly changing and the criminal is constantly starting over on an unsolvable challenge.

The DDKI framework and diva protocol relates to the field of security for electronic communications and in particular network scaling, authentication and Identity Management, detection, revocation and encryption methods.

The most widely used method for providing security online for authentication and encryption is using asymmetrical encryption systems of the public key design where authentication relies on certificates issued by certificate servers. Public Key Infrastructure (PKI) systems have known security vulnerabilities such as being susceptible to Man-in-the- Middle [MiM] attacks, because they are often implemented improperly.

The overhead of the PKI system is high, not just because of all the steps involved in the architecture, but also their choice of cryptography. The encryption strength used by the PKI has been called into question recently. Public keys are compound primes and they are always available for attack. There have been significant strides in prime numbers and factoring theory. New techniques exist to factor compound primes. Fast computers factor compound primes by simplified techniques like the "sieve" method, so what used to take years now can be done in hours. Using progressively stronger keys with public key systems becomes progressively more difficult because of the additional computational overhead introduced as keys get stronger (longer).

There are a number of reasons why security on public key systems is problematic. The Certificate Authority [CA] may not be trustworthy. The private key on a computer may not be protected. It is difficult to revoke keys (refuse network access). Revocation generally requires Third Party intervention. Asymmetric systems are difficult for the average user to understand. Also the cryptographic key information is publicly available to hackers. There are currently no methods of providing continuous, stateful authentication, continuous stateful intrusion detection and automatic denial of network access to hacking and spoofing.

A distributed encryption key is a key that has been pre-distributed by some manual means, such as courier or person to person, to the party involved. This is the most secure method of ensuring key privacy; however this is a problem when new dynamic sessions wish to be established with parties who do not have pre-shared key information.

Any topology or technologies created to provide the highest level of network security must address issues of secure key management, key creation, key exchange, authentication, detection, revocation and authorizations.

IV. NEED

Dynamic Distributed Key architectures as described herein address the aforementioned elements and shortcomings of the PKI system. At the topological level, several network topologies are disclosed that use distributed keys as a random number generator to in turn generate additional distributed keys and securely distribute them to additional devices/persons electronically for easily scalable networks and for scaling secure networks over the Internet. Additionally, these distributed keys can generate session keys for use with any encryption algorithm.

Although the preferred embodiment use new generation authenticated symmetric keys for additional key generation (and for all security functions including encryption), this may be accomplished with any deterministic random (pseudo random) data source and any encryption algorithms. Adoption of secure network topologies also relies in some contexts on its ability to leverage existing technologies.

Secure networks require several components for effective and secure use and deployment. Disclosed are techniques to provide stateful and continuous authentication, detection and automatic revocation. These components are based on the ability to use a deterministic random (pseudorandom) data source to generate and compare portions of a key stream (key output) that have not yet been created and not yet transmitted. Key segments are compared ahead in the key stream. Secure transmission of keys occurs if they are delivered in an encrypted state and an un-authorized party never has access to all the information required to fashion a break or a successful guess of a key stream segment. This also requires the ability to easily manage offsets so each endpoint knows where in the key to begin key stream segment (token) generation.

V. SCALABLE SECURE NETWORKS WITH DISTRIBUTED KEYS DISTRIBUTING MORE DISTRIBUTED KEYS

Effective techniques exploiting these characteristics of Dynamic Distributed Key topologies are provided to prevent Man-in-the-Middle attacks, provide continuous authentication and detection, and safeguard with automatic revocation. This invention uses a distributed key, not as a key for a point-to-point link, as would traditionally be done, but instead that key is used to distribute encrypted "session" keys to be used for the original intention of establishing secure links of communication. Distributed keys allow for the encryption of traffic and but also the authentication of the other party.

The system creates a dynamic distributed key environment that can be used for TCP/UDP tunneling. The Gatekeeper creates and encrypts tunnels based on simple standard netfilter rules, while the Key Vault facilitates the retrieval of point-to-point keys as required by GateKeepers as they talk to each other. In short, the system currently facilitates near-transparent, dynamic, encrypted point-to-point communication between networks on a network. The Key Vault and GateKeeper systems work together to create a layer on any IP based network, like the Internet, that allows communications to remain secure and confidential.

The framework provides a dynamic distributed key system. Traditionally distributed key systems require that a key be delivered through courier or in person to each person with whom one wishes to establish a secure link. This protocol overcomes this encumbrance. At any time, one can start communicating to someone else that uses the invention without having to wait for a distributed key to be delivered.

The framework therefore provides a method of encrypting a communication between a first source computer and a second destination computer, wherein the source and destination computers are each provided respectively with first and second private distributed keys, each associated with a first and second unique private key identifier, wherein a key storage server is provided with the first and second private distributed keys, each associated with the first and second unique private key identifiers, the method comprising: i) the source computer sending a request to the key storage server for a session key; ii) the key storage server identifying the source computer and locating its associated private distributed key; iii) the key storage server generating a unique session key for the session in question, identified by a unique session identifier; iv) the key storage server encrypting the session key with the source computer private distributed key and sending it, with a session identifier, to the source computer; v) the source computer using the source computer private distributed key to decrypt the session key and using the session key to encrypt the communication, which is sent to the destination computer along with the session identifier; vi) the destination computer receives the encrypted communication and session identifier and sending a request to the key storage server for the session key associated with the session identifier; vii) the key storage server determining from the session identifier whether it has the corresponding session key, and whether it has the destination computer's private distributed key; viii) if the key storage server determines from the session identifier that it has the corresponding session key, and has the destination computer's private distributed key, the key storage server encrypting the session key said destination computer's private distributed key and communicating it to the destination computer; ix) the destination computer then decrypting the session key using its private distributed key and decrypting the communication using the decrypted session key.

VI. Description

In what follows, the two components of the system referred to as GateKeeper and KeyVault. GateKeeper is the point to point data link layer tunneling system which uses KeyVault. KeyVault provides keys to GateKeepers as they request them.

The GateKeeper and Key Vault servers can be used in any tier of network architectures traveling from IP to IP, whether from computer to computer, or alternatively, from network to network, or computer to network, and wired-to- wired, wireless-to-wired, and wireless-to- wireless. The system is able to plug anywhere into a network because the system relies on the data link layer between systems. Some other encryption systems rely on the application level (SSH is an example of this). When the application level is used, the secure tunnel is application specific and needs to be re-integrated with each application that wishes to utilize it such as VOIP, e-mail, or web surfing. Using the data link layer instead, allows immediate integration with every IP based application with no delay. The applications do not know that the tunnel is there.

The KeyVault and the GateKeeper applications can work separately, or as a combination. The GateKeeper tunneling system can be used on its own to only facilitate the traditional notion of static point-to-point tunnels that would be useful for ISPs, governments, embassies, or corporations. The KeyVault architecture to distribute session keys based on a distributed key allowing for point-to-point dynamic connections can be applied on other areas apart from the tunnel. These other areas include cell phones to secure calls; e-mail systems to secure and authenticate e- mails; satellites for military satellite image streaming; peer-to-peer networks like Bit Torrent (many ISPs filter peer-to-peer network traffic and give users a slower throughput on those connections; encrypted traffic however cannot be analyzed).

Each GateKeeper workstation has a unique key -pairing with its Key Vault. The two GateKeepers request a session key from the KeyVault using their assigned keys which are assigned physically on installation. They can then communicate with each other using that session key. No single GateKeeper can decrypt arbitrary data. When encrypted data needs to be decrypted, only the destination computer can decrypt it, since only the two computers involved in the transmission can obtain the session keys from the KeyVault since the session keys are encrypted by a unique key pairing with the KeyVault.

The GateKeeper client creates and encrypts the request for the session key with the other GateKeeper with its private distributed key that only the Key Vault that holds the session key has a copy of. Only the two GateKeepers involved in the session can request the session key, as their private keys authenticate their requests with the KeyVault.

The sequences of events that drive a secure link start with the GateKeeper on the initiating side, move on to the KeyVault, and finally end at the receiving side. In both the GateKeeper and the KeyVault, the two systems work together to form the distributed key system in establishing secure point-to-point communication. The GateKeeper communicates through tunnels to other GateKeepers using existing cached keys, and retrieves any needed session keys from the KeyVault as needed. The KeyVault simply receives and respond to key requests.

A source Gatekeeper has a private distributed key which is associated with its unique identifier and stored at the KeyVault in connection with that identifier. See Illustration 1. To commence an encrypted communication with Gatekeeper 23, Gatekeeper 21 sends a request to KeyVault 25 for a session key to. KeyVault 25. KeyVault 25 identifies the sending GateKeeper 21 and locates its associated distributed Key 1. It then generates a unique session key for the session in question, identified by a unique session identifier. It then encrypts the session key with Key 1 and sends it, with the session identifier, to Gatekeeper 21. The source gatekeeper 21 then uses Key 1 to decrypt the session key and uses the session key to encrypt the communication, which is sent to Gatekeeper 23. Gatekeeper 23 receives the packet and determines whether it requires decryption. If it does, it communicates a request to KeyVault 25 for the session key. KeyVault 25 determines from the session identifier whether it has the corresponding session key, and whether it has GateKeeper 23 's distributed key 2. If it does, it encrypts the session key using Key 2 and communicates it to GateKeeper 23. GateKeeper 23 then decrypts the session key using its distributed Key 2 and decrypts the communication from GateKeeper 21 using the decrypted session key.

The Gatekeeper application may consist of one or more pipes, each pipe consists of an incoming and outgoing packet conveyor that is responsible for filtering and encrypting the packets based on the rules from the rule manager in their packet processor, retrieving keys as necessary through the key manager. The Key- Vault application has one main loop that listens for incoming key requests, and fulfills the requests with key responses.

Not only does the system create a secure point-to-point communications layer, but it also provides a way for dynamically adding new GateKeepers to the system without having to copy the key manually to every other client before communication can commence. At the same time it is satisfying the authentication requirement. The problem with SSH (an alternative secure tunnel system) for example, is that it is vulnerable to man-in-the-middle attacks. Distributed keys, by their very nature destroy the possibility of a MITM attack; since, an unencrypted key exchange never occurs, there is never a chance for a hacker to intercept or spoof the keys.

Scalable, authenticated symmetric keys are particularly useful in the present system for several reasons. They are cryptographically strong. They are robust bit-independent encryption. These stream ciphers provide a unique property that most other cryptography methods do not share, that is, once the data is encrypted; the bits are completely independent of one another. This is very useful when dealing with communications because often single bits will get corrupted when transferring large amount of information, and sometimes it is impossible to re-send the information, and so when the cryptography method used fails because of one bit being corrupted, then the data is lost or a huge performance hit is reached due to the necessity to resend the data. This is overcome by being bit independent. If a bit gets corrupted while being encrypted with symmetric, authenticated, symmetric streams the resulting decrypted data is exactly how it would be if it were not encrypted in the first place.

The pre-distributed and pre-authenticated private key is used as AES session key generator thereby eliminating PKI based Trusted Third Parties for session key generation can eliminate this part of server overhead by moving it effectively to the client. Because of its highly random nature and extraordinarily long streams, new generation symmetric ciphers are ideal for this purpose. Key generation can also occur at the server but increases unnecessarily the server overhead.

For Key Generation, the distributed keys (not session keys) are preferably all manufactured using the serial number, MAC#, NAM, or other unique identifiers as a seed in the key generation to manufacture a user/device specific key. This authenticates a device. Only the single device has the correct Universal Identifier to be able to decrypt the device/person specific distributed key with the application key (a secret key associated with the application which is never transmitted and is protected and machine-compiled within the application). This helps avoid piracy and spoofing. Thus to distribute the keys, the server will first send a serial number read utility to a new appliance as a firmware patch. The new appliance sends the MAC#, NAM or UID to the server. The server then generates unique keys and unique starting offsets from the serial number, updates itself with the UID, offset and key information, encrypts the private key with the application key and sends a package with encrypted private key(s) and secure application to the new device.

Packet Authentication Pad may be added to the custom header. This may be used to protect against the possibility that small predictable rejection responses of a server may be blocked and intercepted by a hacker in order to reverse engineer small portions of the stream. This authentication pad consists of another segment of the stream interacting with a CRC checker (which eliminates the possibility of a 100% predictable packet).

IP Fragmentation Completion may be provided. Currently the GateKeeper Tunnel Packet Fragmentation causes approximately a 1 % corruption of fragmented packets. This should be corrected in the system if 100% transparency is to be maintained. This fragmentation is necessary for maintaining packets under the maximum transmission size for Ethernet of 1500 bytes.

The MAC address and IP addresses inside the tunnel may be replaced by the tunnel packet's MAC and IP in the unwrapped packet. This is necessary to ensure compatibility with subnets across the Internet, so the system will work beyond just a LAN or on an exposed Internet connection with no network address translation. A MAC to IP address binding can be added as a failsafe to double- check the authenticity and watch for attack attempts.

Implementing a Key Vault protocol to handle Key Fragmentation will allow the system to handle maximum key sizes of greater than 216.

GateKeeper registration and update management can also be incorporated. This can also be used to add IP addresses dynamically to the list of secure systems so that rules need not be created manually. A logging facility that watches for attack attempts or offset synchronization issues can be added for system administrators to identify malicious activity.

Offset Overlap Checking can be added to see if an offset is being used twice. One can compare the actual data represented by the offsets or the offsets themselves. A pad should never be used more than once; otherwise it is subject to statistical analysis attacks.

Since the system relies on Berkeley packet filter type expressions to determine the types of packets read, this system can be easily integrated with firewall features. Disabling non-encrypted traffic is an option in the GateKeeper system.

The method where the pre-distributed and pre-authenticated private key is used as AES session key generator, thereby eliminating PKI- based Trusted Third Parties for session key generation and eliminating this part of server overhead by moving it effectively to the client. Because of its highly random nature and extraordinarily long streams, new generation authenticated symmetric keys are useful for this purpose. Other Random Number Generators can also be used. Key generation can also occur at the server but increases unnecessarily the server overhead.

First the System administrator distributes a unique private Identity Management key pair on a USB flash memory stick (or other media) to an employee. Alternatively, at manufacturing, devices can have a unique private key associated with a unique device identifier burned into the device during the manufacturing process. The user is authenticated by two factors: possession of the distributed key and a robust .NET password. This process has eliminated the need for a third party authentication.

To send a secure file, the distributed key acts as a random number generator and produces a session key and initialization vectors. Session keys can be any size. This session key generation is done at/by the client and this eliminates any outside Trusted Third Party for session keys. Session key generation can also be done at the server but increases overhead with the generation and secure transmission back to the client. This session key then encrypts the file using a standardized AES encryption algorithmic technique. The encryption process in this manner makes the system AES compliant.

To enhance key security, when the application is initiated the application key uses the unique serial number on the device to decrypt the Private Key. The application will be able to decrypt and use the private key if the serial number is correct.

After having encrypted the file, the session key itself is encrypted (along with initialization vectors etc.) by the sender's pre-distributed AES key contained on the distributed flash memory private keys. The AES encrypted - AES session key is then encrypted again with the distributed authentication key and embedded in the header of the encrypted file. Encapsulating the AES encrypted-AES session key acts as the Identity Management authenticator and strengthens the protection of the session key by adding this strong authentication. A pre- distributed pre-authenticated AES key can also do the second layer of authentication encryption.

This file is sent to the receiver via the framework server/key vault that contains a duplicate copy of all distributed key pairs. At the server, the server's copy of the sender's private key decrypts the encrypted header session key, removing the encapsulating layer of authentication encryption. The server trans-encrypts the session key from being encrypted in the Sender's AES key to the Receiver's AES key. This trans-encrypted session key is then encrypted with the receiver's distributed key, again encapsulating the encrypted session key and being the authentication layer. It is embedded in the header. The file is sent to the receiver.

The receiver is authenticated by having the matching distributed key and by knowing the password to activate it. The receiver is then able to decrypt the encapsulating authenticating layer. This leaves the AES encrypted-AES session key. This is decrypted with the receiver's distributed AES private key. The authenticated and decrypted session key is then used to decrypt the document or file.

The key Identity Management and data protection system has a copy of all physically distributed keys and key pairs for each person/device on the system. The key pairs can be any encryption key pairs. The server may have session key generation capacity for creating new key pairs for physical distribution or for encrypted distribution in a dynamic distributed key environment; or, pre-manufactured key pairs can manually be inserted for availability by the authentication and key vault server for additional security and lower processing effort by the server. In a dynamic distributed key environment, new keys are encrypted and delivered to new nodes encrypted in keys that have already been distributed. This eliminates session key distribution using asymmetric handshaking techniques like Diffie-Hellman. Additionally, this model eliminates the need for Trusted Third Parties (outside sources) for the creation and issuance of session keys. Session key generation, when required, is preferably done by the client thereby eliminating this function as a source of increased server overhead. Session key generation may also be done by the server or outside the server by a systems administrator.

AES session key generation is ideally done at the client preferably using a pre-distributed, pre-authenticated key as a robust, fast, low overhead random number generator to generate AES keys. Dynamic distributed key architectures authenticate pre-qualified users based on something they have (pre-distributed private keys on devices, flash memory etc.) and something they know. This eliminates the dependency on third party Certificate Authorities currently required to establish identity electronically.

In dynamic distributed key architectures, the server can use its ability to trans-encrypt t the secure traffic through the server from being encrypted in the key of the sender into being encrypted in the key of the receiver. Because of the speed, it is possible to transcript the entire transmission (file, session keys and vectors) without negative impact on performance. A preferred alternative, to further minimize the computational overhead at the server when using AES key pairs alone (particularly) is to simply trans-encrypt the double encrypted session key itself.

The trans-encryption process for session keys is as follows. An AES session key is created (preferably at the client). This session key is used to encrypt a file utilizing a standard AES algorithm. This created session key is encrypted with the client's pre-distributed AES private key. This AES encrypted session key is then double encrypted with the pre- distributed authentication key (the other key in the distributed key pair) effectively encapsulating and double encrypting the session key and increasing by orders of magnitude the effective security and bit strength of the protection. At the server, the trans-encryption process authenticates the sender by being able to decrypt the authentication layer with a copy of the sender's distributed authentication key, then decrypting the AES session key with a copy of the sender's distributed AES key, then re-encrypting the session key with a copy of the receiver's predistributed AES private key, and finally encrypting all of the above with a copy of the receiver's predistributed authentication key. The double encrypted session key is then embedded in the header of the file and the file is forwarded to the recipient.

While this is a four-step trans-encryption process, server processing is minimal because only the AES session key is trans-encrypted. For example: a 128-bit AES session key is 16 characters or bytes long. The entire trans-encryption process is only manipulating a total of (16 bytes X 4 steps) 64 bytes. This is negligible even for strong AES keys. It ensures robust security by strong protection of the session key (never transmitted unencrypted electronically) with minimal server processing.

It allows immediate compliance with existing standards while facilitating the gradual transition to stronger encryption and authentication algorithms and techniques.

VII. Dynamic Identity Verification and authentication

The fundamental characteristic of Dynamic Identity Verification and Authorization and the different functions it serves is the ability to generate and compare tokens (key segments) that have never yet been created or transmitted. These and other similar DIVA techniques are ideal for identity verification, history logging and deniability or non-repudiation, Internet based secure payment topologies and secure site access, SCADA topologies etc. (but not restricted to that).

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 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 the Server acknowledges by sending authorization. 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 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 Ik 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 card holder uses their card the transaction is refused because the stolen card 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. 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

D. Authorization/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.

There are many obvious topological configurations possible by changing where the different components of key creation and storage, authentication, detection and revocation occur between a client, server, person, device or a proxy. Individual components may be used in other network topologies for additional layers of security abstraction.

All technologies described in this paper are patented and require licenses for commercial use.

VIII. DIGITAL IDENTITY AND TURNING BIOMETRICS INTO A ONE-TIME-PAD

Abstract— Currently if a person's biometric data is stolen or hacked or broken their identity is potentially compromised for their entire life. Your biology will never change.

A major handset maker released a phone with a fingerprint scanner in 2013 and it was publicly broken within two days in Germany. This biometric is a major component of a mobile payment solution recently rolled out.

Using a Whitenoise key and Dynamic Identity Verification and Authentication (DIVA), one can deterministically turn a biometric into a one time pad.

IX. INTRODUCTION

We will first look at how a Whitenoise key is constructed. Then we will then look at how Dynamic Identity Verification and Authentication works and why it operates as a one-time-pad. We will then look at how to deploy it with biometrics to create a biometric one-time-pad.

This paper also provides an opportunity to look at a comparative bench mark for Whitenoise and scientific curiosities that merit additional research. And, most importantly to look at legal and moral implications of cryptography as additional technical capabilities are deployed.

X. CREATING A WHITENOISE KEY

Whitenoise is a deterministic random number generator. It creates key streams that appear to be orders of magnitude more random then a sample of radio-active decay. One key creates an infinite number of one-time-pads and can handle all your network security controls.

A Whitenoise keys is built from a variable number of prime number length sub-keys which create the data source. Each bit is XOr'd with the corresponding bit of the next subkey to create the first key stream. To delinearize this stream, two bytes worth are appended together and run through an S-box and only one byte emerges. That becomes first byte of the delinearized key stream which can then be used cryptographically.

This creates several layers in its one-way function. 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 are the only provably unbreakable key construct and 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."

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.

XI. 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 (key segments) that have never yet been created or transmitted.

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 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 the Server acknowledges by sending authorization. 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 card holder uses their card the transaction is refused because the stolen card 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. 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.

XII. Apply DIVA to Biometrics (any kind)

By deterministically randomizing the coordinates of a biometric authentication call, you can turn the biometric into a one-time-pad. You can lower the number of coordinates compared for biometric authentication and still achieve a significantly highly higher level of security with lower overhead. This small footprint is ideal for mobiles.

With Whitenoise if a person's biometric is cracked or stolen, then all that is required is to change the dynamic offsets to the Whitenoise-DIVA key that has been bound to it. That cannot be guessed or stolen. The person's biometric has been stolen but the association with the binding of a Whitenoise distributed key allows dynamic changing of the biometric authentication coordinates making its theft irrelevant.

Iris biometrics is a preferred kind of biometric because they are unique, cannot be forgotten by a user and cannot be stolen. Most computers and mobile devices are already equipped with cameras that can perform this iris scan with the proper accompanying utilities. It will be fairly simple to address blue iris spectrum camera needs.

Dynamic Distributed Key Infrastructures (DDKI), DIVA and iris biometrics (or any biometrics) are distributed key systems and protocols where both the endpoint and the server has a copy of the key. They provide authentication for assured identity and network access (iris identification). DDKI lays down a dynamic distributed framework; DIVA provides complete network and transactional security; and, iris authentication binds a biometric key to a digital key.

The first authentication happens with iris authentication to then move onto the DIVA identification, authentication call and binding of the biometric to the digital key for the session.

Successful authentication with an iris biometric and device and DIVA authentication will allow access to the account and the secure, unique DIVA key within the database. This marries ISO/ITU Identity Proofing Level 4 for human endpoints with a unique, digital DIVA Identity Management key which provides distributed identity and ensures all digital network and transactional security (as well as logs of all use) for that authenticated person throughout the life of the session.

A biometric binding organic identity to a digital DIVA key ensures secure, tamper-proof network access and continuous, in-session, dynamic authentication.

No one buys a home alarm until their house has been robbed. Security should be inherent in any service we use. Public security measures for persons often require some kind of invasive touching or something that many consider offensive or intrusive i.e. scanning genitals. Most identification requires that a physical key is carried with the person.

EVERYONE, everywhere, understands the uniqueness of their own biometrics. A user cannot forget or give away his key. It is always with them. It can't be tampered with. A user does not require any other physical key for identity based network access – no USB drive, no credit card, no health card, etc.

Two completely unique keys for network security (a person's iris and DIVA) bind organic identity to digital identity using ITU/ISO Level 4 identity proofing.

A person does not need to be touched. It is not intrusive. It is legal in all cultures and societies (developed, underdeveloped, literate, and non-literate.) About one third of the world is Christian. About one third of the world is Muslim. About one third of the world is Buddhist, Hindi or something else.

This approach works in and is legal in any culture or society. The iris biometric is a totally unique distributed key. Successful authentication by iris scan authorizes the person to use an account with a unique, identity-based, dynamic, distributed digital DIVA key (organic to digital).

The DIVA key is secure within the application and can't be accessed and tampered with. Authorized use of the DIVA key then provides complete internal and external network security.

This approach acknowledges and honors various religious and cultural contexts while providing complete security. In conservative or religious cultures that require a dress code, an iris scan is a reasonable requirement that can be demanded legally. No touching. No photographing genitals. Nothing invasive. Discreet.

An iris scan is a reasonable security identifier globally. Many countries are experiencing rapidly growing, diverse ethnic populations. Global travel, tourism, business, migrations etc. create larger, more diverse, flowing populations that need to be identifiable to access public and commercial services in digital contexts. This approach is functional in the most trying of contexts i.e. identifying persons for distribution of humanitarian aid after disasters.

Any digital service is simple and secure: securing banking, moving through an airport, or traveling through a foreign country etc.

Dynamic Distributed Key Infrastructures DDKI and Dynamic Identity Verification and Authentication DIVA pick up from there and satisfy all the ITU/ISO criteria for securing data, communications and networks.

DIVA catches all false positives and false negatives to make biometrics 100% accurate while securing all transactions.

A person distributes a scan of a biometric (fingerprint, face scan etc.) to the server one time. A scanner takes a "snapshot" and compares specific co-ordinates against the stored copy.

The more points compared, the greater the accuracy and fewer false positives – but the greater the cost. Mass market biometrics compares fewer points but have more false positives. This defeats the purpose.

Note: DIVA and Whitenoise can be used to randomize the coordinates that are compared between an end-point scanner and minimize the number of coordinates that need to be compared (because it is now operating like a one-time-pad) in order to get an acceptable level of assurance while minimizing the attendant costs of utilizing biometric information.

An iris biometric binds identity to a diva account. This can be done with 100% accuracy with digital keys using ISO/ITU Level 4 assurance which requires that identity proofing be local (same as biometrics) or electronic. The biometric binds identity to a diva key which in turn can be used for complete network security and identity management. The diva key can also be used to secure and store the biometric 'key' as well.

XIII. Applying One-Time Pad Biometrics to Scalable, Secure, Adaptive Mobile Networks

It is not daunting to either fix or harmonize all network communications. All components of the network are identified by a unique key. All persons/devices are identified dynamically and continuously. All usage is logged.

For perfect identity management and security a device only needs a little bit of storage space for keys, write back capacity to update dynamic offsets, and an internet connection or connectivity.

Dynamic identity verification and authentication [DIVA] is an identity-based, software protocol that can be used in any digital context that addresses all security requirements: dynamic and continuous, authentication, authorization, revocation, inherent intrusion detection, digital rights management, digital signature, and secure network access.

Users are pre-authenticated and keys are pre-distributed. Distributed keys eliminate PKI attacks. It provides end-to-end, hop-by-hop, endpoint-to-node or node-to-node authentication.
DIVA operates as a one-time pad. It can provide perfect identity for persons and devices. It can also be used for pseudo-identity and anonymity in other contexts.

DDKI/DIVA satisfies all ITU/ISO requirements for Identity Management and Privacy by Design protocols. It operates on any kind of digital network and any computer operating system like Windows. It can be used in the following environments: federated, silo, centralized, user-centric, or mobile centric.

It can be used with PKI or in lieu of PKI.DIVA and DDKI provide a completely interoperable and scalable software framework that isn't hardware dependent.

PKI and DDKI can happily live together and compliment one another. DIVA fixes all the fatal flaws of PKI.

DIVA and DDKI work seamlessly with Public Key Infrastructures without direct integration into any of your existing security controls or frameworks.

DIVA and DDKI in conjunction with PKI (the predominantly implemented network security scheme) raises the bar by creating a two channel (both asymmetric and symmetric frameworks) multi-factor authentication protocol.
The attacker then needs to break keys from two unrelated frameworks, one of them dynamic (DIVA), for each and every breach. You can see in the DEFCON – Black Hat Challenge the key in this example is set to dynamically change every 15 seconds.

Everyone needs a technology shift without the disruption. DIVA and DDKI accomplish that without threat to any network in transition. Note: there will not be a single, secure PKI network on earth within five years when quantum computing arrives because of the fixed keys sizes.

It is accepted now that keys from existing technologies (public keys) can factored and broken just with existing computing speeds and brute attacks. There are also mathematical attacks. WN cannot be factored or broken.

Integrate DIVA into a Single-Sign-On login protocol for network or application access. This plugs the fatal flaws of PKI which is man-in-the-middle attacks and side channel attacks. PKI is safely transitioned.

First add DIVA to protect the network and augment existing security. (i.e. surround the oil slick and leaks first!) Over time remove expensive, redundant or ineffective existing security components.

PKI was an ad hoc approach implemented after the fact. PKI was never scientifically qualified to be a ubiquitous, completely secure framework. PKI is ALWAYS vulnerable to man-in-the-middle attacks. PKI is ALWAYS vulnerable to side channel attack classes. After 40 years < 10% of North American use enterprise PKI servers. After all this time why it is still so vulnerable and why aren't all the markets saturated?

We need fundamental, safe, shift to include DIVA and DDKI into our cyber defenses. PKI will be used in limited, specific contexts that DIVA can secure because a key cannot be stolen or copied without being detected when DIVA is present.

Even if public key systems could provide an equivalent level of security as dynamic distributed key frameworks, DDKI and DIVA provide the lowest cost, the simplest remote provisioning, installation and enrollment, the simplest security to understand, train and manage, simplest framework to configure for international commerce while satisfying needs of different countries to control their own national security and dialing in their own unique approach.

Managed mobility services (MMS) encompass the IT and process services provided by an external service provider (ESP) that are required to: plan, procure, provision, activate, manage and support mobile devices, network services and mobile applications.

The major design goals of the architecture are:

- mobility as the norm with dynamic host and network mobility at scale
- robustness with respect to intrinsic properties of the wireless medium
- trustworthiness in the form of enhanced security and privacy
- usability features such as support for context-aware services,
- evolvability, manageability and economic viability

Achieving MMS is simple with DDKI systems and DIVA. DIVA keys can be kept at one location; addressing can be dynamic and handled by separate DCHP servers. Dynamic offsets can be kept at yet another location. These are simply design decisions.

Certificateless authentication with a distributed key infrastructure is a better option in terms of speed, overhead, unbreakable identity etc. then public key certificates. But, should a designer want to work with PKI for additional authentication the two systems work together seamlessly to create a two channel, multifactor authentication challenge.

In delay tolerant networking DIVA and DDKI are excellent choices. In this tunnel paradigm with a Key Server and Gatekeepers both WN-DIVA keys, and their offsets, and addressing information is appended into packet headers. This allows the simple storage of packets in DTNs until connectivity is established and then normal routing would continue.

It supports packet switching and can support hop-by-hop transport in a store and forward manner.

DIVA and DDKI can work with any security controls desired. Please study the tunnel paradigm below: http://www.wnlabs.com/Tunnel_Distributed_Keys_distributing_more_keys.pdf

Dynamic Distributed Key Infrastructures (DDKI) is a virtual, distributed, tiered, hierarchical network of secure networks of devices deploying Dynamic Identity Verification and Authentication (DIVA). One distributed key will create an infinite number of one-time-pads and a distributed key can in turn distribute more distributed keys.

Predictive, preventive, detective and response capabilities are all characteristics of Dynamic Identity Verification and Authentication (DIVA – a protocol) and Dynamic Distributed Key Infrastructures (DDKI – a virtual frame work). Predictive includes being able to determining where failed hacking attempts are occurring from. Predictive heuristic anomaly detection is easily added.

DIVA and DDKI are INCREASINGLY effective against advanced attacks:

DIVA operates as a dynamic one-time-pad where a single key can create an infinite number of one-time pads, the only mathematically proven key technology. There is a copy of a key and its last dynamic offset at both the endpoint and the server. The keys are either synchronized or not. The dynamic offsets have to be identical. If not, the system automatically disables the account without human intervention.

- 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
- Botnet attacks are prevented by configuration with server so the botnet never has access to the entire key and offset information for outbound traffic.
- Quantum computing attacks are prevented because every variable is variable
- Brute force attacks are not feasible

XIV. A SPEED BENCH MARK

Dr. Rivest, one of the founders of RSA, recently discussed a paper at the Charles River Crypto Day 2014 on the following subject:

"We estimate that Spritz can produce output with about 24 cycles/byte of computation. Furthermore, our statistical tests suggest that about 2^{81} bytes of output are needed before one can reasonably distinguish Spritz output from random output; this is a marked improvement over RC4."

Alternatively, Whitenoise can produce output with about 2 bytes per clock cycle computation. And that is scalable. That is orders of magnitude faster than Spritz with virtually no overhead or computational requirements. Whitenoise security technologies are unique is that they are just as fast in software as we are in hardware.

This illustrates exactly why traditional, current technologies cannot effectively deploy even 128 bit keys in the majority of devices that comprise the Internet of Things and the Cloud of Things. Internet of Everything (IoE) is a $19 trillion dollar market.

Because of computation effort of asymmetric systems they are not feasible in the majority of the products comprising the Internet of Things are characterized by low cost, storage, power and computational processing ability. No one is putting a $10 chip set in a $30 product.

Conversely Whitenoise technologies are easily deployed in this environment and are the only national security level crypto that can effectively be deployed in "Peripheral Interface Controller (PIC), the cheapest microprocessors available.

XV. SCIENCE CURIOSITIES ABOUT WHITENOISE TECHNOLOGIES THAT MERIT FURTHER INVESTIGATION

There is not supposed to be random anything, only pseudo random. This is how it is characterized in cryptographic sciences.

In the performance analysis conducted by the University of Victoria ECE department, a super computer array was constructed and Whitenoise was tested against the NIST test suite. This randomness test suite allows for one statistical error for every hundred rounds. This suite was made more sensitive to allow for only one statistical error for every thousand rounds.

In weeks of testing there was not even a single statistical error.

http://www.wnlabs.com/downloads/UVIC_Performance_Analysis.pdf

It is not generally accepted that it is possible to add entropy to a system. When constructing the data source for Whitenoise keys multi subkeys of prime number lengths are used.

It has been reported that entropy increases as more subkeys are added.

Key generation utility and speed tester:

http://www.wnlabs.com/downloads/WNspeedUtilitydemonstrator.zip

These are patented and are available only for academic, non-commercial use. Commercial use requires a license.

Randomness testing suite:
http://www.fourmilab.ch/hotbits

XVI. CAPABILITIES OF USING THESE SYSTEMS TO PREVENT ENTRY OF TERRORIST RECRUITMENT VIDEOS THAT INSPIRE LONE WOLF TERRORIST ATTACKS

All crypto technologies are designed to attempt to secure data. Dynamic Distributed Key Infrastructures and Dynamic Identity Verification and Authentication can also be used to regulate data.

Raising this question does not presume to say what should or should not be done – only that it can be done. The legal implications to free speech and democracies in the digital age are ones that should be subject to constant and diligent scrutiny. One of the challenges of our times is balancing security and privacy.

The question has been posed: Should terror recruitment videos that inspire lone-wolf attacks be allowed to come into sovereign territory unregulated?

Free speech is regulated – you can't yell fire in a movie theatre.

Al Capone was eventually controlled by tax laws.

We can look at data on the Internet as digital goods. And goods fall under the mandated responsibilities of customs.

It is possible to create Dynamic Distributed Key systems where there is a finite and manageable set of servers that act as gatekeepers and ports of entry for digital data entering our country.

At this point, we can apply regulations, as we already do, for goods entering our country. Unidentified goods without identity, keys, or authorization are instantly recognized at these points. They can be allowed or refused.

Applying this capacity to digital data to try to achieve the above goal which many espouse brings up an array of questions dealing with free speech and censorship.

Finding a balance will be fundamental to balancing security and privacy in democratic countries. The answers should be arrived at by democratic means.

All technologies described above are patented and require licenses for commercial use.

XVII. A WHITENOISE POINT OF VIEW ON FIC HOT TOPICS

A. How to take down a botnet?

Whitenoise takes a different approach. We aim to prevent them in the first place.

Criminals don't want to identify themselves. They commandeer remote computers. From these computers they send malware that gets installed on victim computers and networks. They steal important data and then they SEND this data back to the botnet computers. If they cannot send stolen data back out to themselves to use for criminal purposes then no harm is done by the botnet. The malware is isolated on a computer and can be tracked down and removed.

By configuration, Whitenoise technologies combat botnets by requiring authentication and provenance of data go OUT of a computer. There are at least two keys, or offsets, with one that reside outside (on another computer). The botnet can never get access to the full key material required to send stolen data out.

If all persons, components and devices have requisite identity on secure networks, then the players we are looking for are operating on unvetted and unsecured networks. This is where we need to fix our gaze.

B. Illegal content from detection to removal?

This challenge brings up critical questions to be collectively answered in the balancing of security and privacy. To do so we will look at one specific kind of case and capability.

1) Using cyber technologies to prevent entry of terrorist recruitment videos that inspire lone wolf terrorist attacks that we are witnessing on an almost daily basis.

All crypto technologies are designed to attempt to secure data in transmission and rest, and to assign and enforce provenance on the data. Dynamic Distributed Key Infrastructures and Dynamic Identity Verification and Authentication can also be used to regulate data.

Raising this question does not presume to say what should or should not be done – only that it can be done. The legal implications to free speech and democracies in the digital age are ones that should be subject to constant and diligent scrutiny. One of the challenges of our times is balancing security and privacy.

The question has been posed: Should terror recruitment videos that inspire lone-wolf attacks be allowed to come into sovereign territory unregulated?

Free speech is regulated – you can't yell fire in a movie theatre.

Al Capone was eventually controlled by tax laws. We can look at data on the Internet as digital goods. And goods fall under the mandated responsibilities of customs.

It is possible to create Dynamic Distributed Key systems where there is a finite and manageable set of servers/towers that act as chokepoints, gatekeepers and ports of entry for digital data entering our country.

At this point, we can apply regulations, as we already do, for goods entering our country. Unidentified goods without identity, keys, or authorization are instantly recognized at these points. They can be allowed or refused entry following legislated regulations and process.

Applying this capacity to digital data to try to achieve the above goal, which many espouse, brings up an array of questions dealing with free speech, censorship, competition etc.

C. New technologies: what security challenges?

First the technologies must exist and thankfully they do. The ultimate challenge (besides overcoming learned helplessness and fostering collaboration) is that technologies must have a transition and implementation roadmap that will augment and harden existing cyber security systems (predominantly Public Key Infrastructures) before removing components that are ineffective and redundant. No one is leaving a nuclear envelop or banking system unguarded for even a nano-second even if the security isn't the best. Surround the oil slick before it spreads more, and then clean up.

Whitenoise technologies, a secure virtual network framework and virtual protocol, are designed to work seamlessly without direct integration with any other framework, security controls and topologies. They are designed to run in parallel and are usually invoked at the point of secure network access.

D. Smart City (electrical grids to thermostats)

Smart Cities will enable us to be as green as possible. Smart Cities are also inevitable and desperately needed particularly for the use and sharing of finite and limited resources like energy. The sharing of electricity between California, United States and British Columbia (BC), Canada is a good example.
Peak use of electricity happens at different parts of the year for each place. During the winter, BC keeps most of its electricity when it is needed for heating. During the summer, instead of

dumping excess production, electricity is sent to California that needs more energy to run their air conditioners.

Smart Cities like the Internet of Things, Internet of Everything, and the Internet of the Cloud have a growing and unique set of vulnerabilities. The majority of connections are machine-to-machine with limited or difficult human oversight. And, a huge percentage of the components will be low cost, low memory, low power devices with connectivity and therefore a significant point of security vulnerability. Without attempting to mix subjects, but rather looking at Smart Cities as having an area overlap with the Internet of Things, we can see the scope of danger. Hewlett Packard recently released a study indicating that there is an average of 25 security flaws in the average component making up similar systems:

http://www.telegraph.co.uk/technology/internet-security/11000013/Average-Internet-of-Things-device-has-25-security-flaws.html

http://www.wnlabs.com/pdf/Internet_of_Things_and_Whitenoise_Technologies.pdf

Digital identity

In this paper, we have already examined using biometrics to bind organic identity to protected digital identity.

At a global, over arching perspective, it is possible to provide one unique key for each person and device that will be protectable identity for their entire lives.

It is critical for cooperating countries to develop a national strategy for identity management that will support Organization of Economic Cooperation and Development goals in order to facilitate secure online e-commerce, and e-government and social networking. The paper Harmonizing Strategies and Policies For Identity Management Whitenoise Laboratories (Canada) Inc. Vision was filed at the First US National Cyber Leap Year Summit and at the United Nations International Telecommunications Union.

http://www.wnlabs.com/papers/National_Leap_Year_Summit_The_Whitenoise_Vision.pdf

http://www.wnlabs.com/papers/Game_Change_Digital_Provenance_post_conference.pdf

http://www.wnlabs.com/Papers/generic_idm_policies-for-international-harmonization_whitenoise.pdf

http://www.itu.int/oth/T1508000003/en

How to improve code security?

Whitenoise technologies are data agnositic. The security is effective because it imposes identity and provenance on all data. Code data is just a subset. The technologies discussed in this paper above provide provenance and identity for all access to networks and the data therein.
Because Whitenoise technologies are so flexible, code specific deployments like Strong Crack Protection have been designed and implemented. The technologies discussed in this paper address access to data conceptually from outside of the data in question. Strong Crack Protection is a deployment of technologies from within the data (code) that will authenticate and handle authorizations from within the data (code) to combat theft, unauthorized installation and use, unauthorized duplication, and unauthorized modification of code. This code can also be encrypted.

Preventing and detecting information leakage?

The Whitenoise technologies discussed in this paper addresses provenance and access to data and provides inherent logging of all network use for forensics.

The security of mobile platforms Android vs IoS

The security of mobile platforms in general was a topic was covered in a Gartner Vendor Briefing on securing biometrics and designing secure, adaptive frameworks for Managed Mobility.
http://www.wnlabs.com/downloads/Gartner_Video.mp4

Are alliance possible in cyberspace?

Political and commercial alliances in cyberspace are a must. Politically things like digital embargos might be used effectively. Commercially, the players must prioritize the overall good before their own particular commercial competitive advantages and within their commercial competitive roadmaps. We all will remain vulnerable without a shared, general, cyber security vision.

Commerce and Security

eCommerce and general economic growth will be slowed down or impeded because of poor security. Secure commerce is one of the goals articulated by the Organization of Economic Cooperation and Development and the United Nations International Telecommunications Union. Commercial entities will realize that security is a competitive advantage that can be monetized.

http://www.wnlabs.com/news/UN_ITU.php

http://www.itu.int/dms_pub/itu-t/oth/15/08/T15080000030001PDFE.pdf

Using big data for security

This issue allows the proposition of looking at big data in a different way.

The rationalization for NSA and other agencies globally to collect data on everyone is that this information will be available if a national security situation ever arises that requires access to this information. It would just be available.

It has been questioned as to whether having any of this information has thwarted serious security events. It is also questionable as to whether if such an event happened that they can search through this information fast enough to be of use in time sensitive situations.
The majority of people are decent and law abiding. They have no issue with identifying themselves – criminals do. Citizens have justifiable concerns about the misuse of big data.

If the majority of people identify themselves with protectable digital identity, then law enforcement, military and governments will need to direct the majority of their resources to persons not on secure networks. This will ultimately dramatically reduce the universe of malcontents and criminals they need to watch in order to protect us. And this protects the democratic right to as much privacy as possible for citizens.

The use of big data for commercial reasons is a different issue.

How to secure data in the cloud?

Whitenoise technologies create secure point-to-point, encrypted communications and tunnels through the cloud and create secure virtual networks within the cloud. This was recognized in 2014 in the Nokia Telecom Council of Silicon Valley Open Global Innovation Challenge for the Cloud and Colossal Data.

http://www.wnlabs.com/papers/Nokia_SV_Open_Innovation_Challenge_WNL.pdf

Security of SCADA

SCADA is characterized by a preponderance of communications between machines. Since so many of these critical communications happen M-2-M it is a requirement that any cybersecurity systems are self monitoring and self healing with inherent intrusion detection and automatic revocation capabilities.

What innovative solutions in the field of encryption

One Whitenoise distributed key creates an infinite number of one-time-pads, prevents all know cyber attack classes and provides all network security controls including identity, secure network access, continuous-dynamic authentication, authorization, signature, inherent intrusion detection and automatic revocation.

http://www.wnlabs.com/papers/Whitenoise_Executive_Overview.pdf

Cyber defense as a service

Cyber defense as a service is already here with contractors like Booz Allen providing a host of defense services.

In a broader context, cyber defense as a service can be more effectively enabled by a platform like Whitenoise technologies where citizens, enterprises and governments can go to a location online and effectively create a virtual server or a physical server for their defense needs. The security needs to be effective and stake holder programmers should be able to go to such a site to integrate national security level technologies into their own networks, services and applications. www.wnlabs.com

Cyber security and health?

Proper and secure sharing of information, particularly through the cloud, for the health care field will improve the rate of good medical outcomes. This is true for global health issues like Ebola and our personal health care process. Companies like eGlobalHealth.com are at the leading edge of this capability.

The range of uses is broad as well from the monitoring of health vitals remotely (think pacemaker) to embedding secure patient data within medical imaging like X-Rays to prevent the separation of patient demographics and imagery. The challenges of space, power and processing capacity in many devices are identical to the challenges posed in the Internet of Things.

Incident Management ?

Incident management is inherent within DIVA systems. There is instant recognition of unauthorized access attempts and instant revocation of network access without human intervention. If a breach possible, forensics would know the exact duration of an event beginning at the last point where the legitimate key was synchronized and ending at the point where network access was revoked. This is the forensic universe.

After Snowden how can we restore trust?

Trust can be restored by the public understanding technology that works and by governments being transparent enough to assure its citizens that it is following its own rules. It needs constant democratic involvement in providing oversight and in legislating rules that ensure adherence to accepted process.

In the United States, after the Snowden event, part of the process played out in the following way:

Telecommunication carries and major service providers pointed a finger at government and said that government was forcing them to collect data and turn it over to the government.

The Obama administration tried to balance this situation by suggesting that the telecoms and major service providers collect and store the Meta data in the event that it might be needed in the future.

The telecoms and major service providers balked because they don't want to hold the data and also hold the liability. They want it both ways.

Among the many characteristics of Whitenoise is that it is bit independent. One of the things this means is that keys and data can be parsed and then reassembled. It was suggested to the Office of Science and Technology Policy cyber czar that master keys can be authorized by government for specific providers (as they currently do) and that the keys can be parsed into at least three segments. One portion would be held by the government, one portion would be held by the service provider, and one portion would be held by an third, independent oversight group.

All parties share the same amount of liability or no liability at all.

In the event that a situation arises where there is a probable cause, all three parties would need to be involved in order to reconstitute the keys in cases of court authorized access. No access could be accomplished in isolation or in the dark.

What role for cyber in military?

The role for effective cyber in the military is critical because cyber warfare is already here. Properly constructed networks will enable the ability for digital embargos on rogue players and minimize the necessity for armed intervention in many situations.

We must be able to protect ourselves from cyber warfare. Cyber has pervasive impact throughout the entire military ecosystem from mesh networks, to surveillance streaming, to tracking of assets etc.

Just as for government, it is critical that military uses remain within their legislated parameters and perimeters. A robust and well secured military is requisite in a dangerous world. The effective balancing of security and privacy aims for the assurance that there is no straying from their mandates.

Future of Security

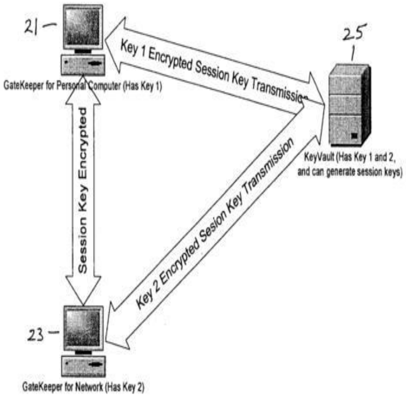
The Future of Security is here. Whitenoise technologies were recognized as a grand finalist in the Future of Security global contest Raytheon and IPSEC sponsored. http://www.wnlabs.com/news/ifsec.php

Whitenoise technologies were also twice recognized as grand finalists in the Global Security Challenges.
http://www.wnlabs.com/news/GSC_Grand_Finalist.php

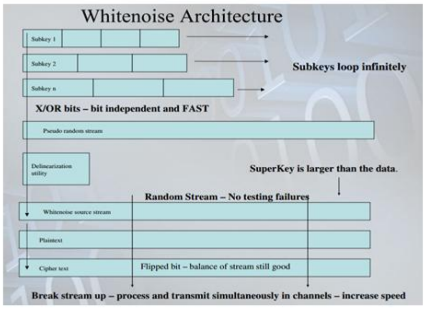
It was recognized as a potential fail safe for quantum crypto by the European Telecommunications Standards Institute:
http://www.wnlabs.com/news/standards.php

Illustrations

A tunnel topology


21
GateKeeper for Personal Computer (Has Key 1)
Key 1 Encrypted Session Key Transmission
25
KeyVault (Has Key 1 and 2, and can generate session keys)
Session Key Encrypted
Key 2 Encrypted Sesion Key Transmission
23
GateKeeper for Network (Has Key 2)

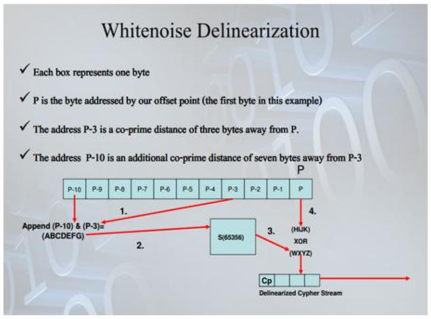
How is a Whitenoise key created?


Whitenoise Architecture

Subkey 1
Subkey 2
Subkey n
Subkeys loop infinitely
X/OR bits – bit independent and FAST
Pseudo random stream
Delinearization utility
SuperKey is larger than the data.
Random Stream – No testing failures
Whitenoise source stream
Plaintext
Cipher text
Flipped bit – balance of stream still good
Break stream up – process and transmit simultaneously in channels – increase speed

- variable number of prime number length subkeys
- each bit is XOr'd with the corresponding bit of the next subkey
- two bytes worth are appended together and run through an S-box
- it becomes first byte of delinearized key stream

What are the Whitenoise one way functions?


Whitenoise Delinearization

✓ Each box represents one byte

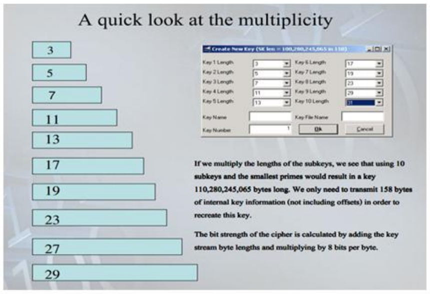
✓ P is the byte addressed by our offset point (the first byte in this example)

✓ The address P-3 is a co-prime distance of three bytes away from P.

✓ The address P-10 is an additional co-prime distance of seven bytes away from P-3

- 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 of key stream from one byte of captured information
- the hacker has no knowledge of the number of sub keys, their lengths or the random data they are populated with
- it is a one-time pad

How are length and strength of Whitenoise keys calculated?


A quick look at the multiplicity

If we multiply the lengths of the subkeys, we see that using 10 subkeys and the smallest primes would result in a key 110,280,245,065 bytes long. We only need to transmit 158 bytes of internal key information (not including offsets) in order to recreate this key.

The bit strength of the cipher is calculated by adding the key stream byte lengths and multiplying by 8 bits per byte.

- the length of a Whitenoise key is calculated by multiplying the length of the subkeys in bytes.
- the strength of a Whitenoise key is calculated by adding the lengths of the subkeys in bytes and multiplying by 8 bits per byte.
- to create a key > 100 billion bytes long, we only have to store 158 bytes of information

How does DIVA work?

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 twenty-five bytes. It is not sending across either an offset or a key with this request.

Last valid offset

Device state 1a

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

The key stream is a minimum of 10^{60} bytes in length. 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 25-byte token beginning at its last valid offset.

Last valid offset plus token

Device state 1b

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 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 the

❑ **Server acknowledges by sending authorization**

❑ **Both server and endpoint update dynamic offset independently**

Last offset

New offset = last offset + token + 1

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.

What are the only DIVA outcomes?

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 the

❑ Server acknowledges by sending authorization

❑ Both server and endpoint update dynamic offset independently

Last offset

New offset = last offset + token + 1

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.

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 the

❑ Server acknowledges by sending authorization

❑ Both server and endpoint update dynamic offset independently

Last offset

New offset = last offset + token + 1

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.

References

[1] Ghodosi et al., "Pseudorandom Sequences obtained from Expansions of Irrational Numbers" presented at CPAC Conference "Cryptography Policy and Algorithms Conference", Queensland University of Technology, Brisbane, Australia, Jul. 3-5, 1995. cited by other .

[2] Carroll, John M., "Do-it-yourself Cryptography", Computers & Security, Elsevier Science Publishers, Amsterdam, NL, vol. 9, No. 7, Nov. 1, 1990, pp. 613-619. cited by other.

Acknowledgments

Albert Meyburgh; SLB

Canadian Cyber Accelerator, Cambridge Innovation Center, Cambridge, Massachusetts, United States of America.