SKDP

The Symmetric Key Distribution Protocol

A scalable, lightweight, high-security symmetric communications protocol.

SKDP

The Symmetric Key Distribution Protocol

Security for a New Age

Key distribution, is one of the most challenging problems in cryptography. The internet has grown at an extraordinary pace since its inception, and is now a core communications medium that is used by billions of people around the globe. The private information we send over this public medium, must be secured, as the internet has now become a primary tool in global commerce, and a communications infrastructure connecting people everywhere. The problem has always been, how do we distribute keys between connected devices, and do it in such a way as to best guarantee the continued security of that service.

The security mechanisms most widely used today, use asymmetric cryptography; public/private key cryptography for encryption and authentication. These asymmetric primitives use 'trapdoor' functions, whereby a part of a difficult mathematical problem is created using a public key, and solved using the private key. The problem with this approach, is that the underlying mathematical problems used by these asymmetric ciphers and signature schemes, are constantly being challenged by new knowledge and advances in computing technology. What seems an intractable problem today, could eventually be reduced, or even solved, at some future time. This is why asymmetric parameters are continually adjusted to make the problem more difficult, and why entire orders of asymmetric cryptography based on large integer factorization and elliptic curves, will soon become obsolete, due to the emergence of quantum computers.

It has been well established, that intelligence agencies collect and store encrypted communications streams on a vast scale. This is because even if the technology to break the underlying encryption technologies does not currently exist, at some future point in time it will, and all of that collected traffic will one day become readable. We could face the same problem with quantum resistant lattice-based cryptography in ten or twenty years, that we face now with cryptography based on elliptic curves or large integer factorization; eventually the technology, and the mathematics, will evolve and combine to create a new threat, a new way to break that cryptographic system. This is further complicated by the choice of parameters used in the design of asymmetric primitives, which are calculated based on projections established only in current knowledge and technology, in a very performance-oriented field, that too often chooses less aggressive parameters in favor of improved performance

characteristics. That we know communications are being captured and stored, but we cannot know what breakthroughs in technology are on the horizon, creates a serious problem that must be addressed. We do not believe that any system based on asymmetric cryptography, can promise true long-term security, which must now be considered as the lifespan of a human being.

Symmetric cryptography may provide a part of the solution. Given sufficiently 'strong' symmetric cryptographic primitives, and longer key lengths, symmetric cryptography is far more computationally expensive to solve, and perhaps even impossible to break for an indefinite time. The problems that have faced systems that use pre-shared symmetric keys, have always been of scalability, and the concentration of security onto a single point of failure. For example, there are systems that use a single pre-shared key and session counter, to key a symmetric cipher and establish an encrypted tunnel, some SSH implementations use this naïve scheme. The problems being, that if the device is ever captured, all past messages will be readable, likewise if the server's key database is captured, all messages, for all hosts on the network, past, present, and future, will be instantly readable by the attacker. There is no forward secrecy, whereby capturing the key for a session, does not reveal anything about past sessions. This scheme is also difficult to scale, and difficult to protect, as the server's key database becomes the focal point of attack.

What we propose with SKDP, is a distributed symmetric scheme that uses pre-shared keys in a way that does provide forward secrecy, that is scalable, and solves many of the problems associated with existing encryption schemes that use pre-shared keys. In SKDP, capture of a client's embedded key, does not compromise past messages, and capturing the server's key does not reveal anything about past encrypted messages, because the symmetric ciphers used in the message stream, are keyed with ephemeral keys, that cannot be derived from the pre-shared key alone. The pre-shared keys are used primarily for authentication and encryption of secret tokens, passed between the server and the client, that cannot be derived from either the key database, or a client's embedded key. The only way to break this system, is to own the master key, and to capture every complete message stream. This is the same vulnerability in asymmetric based schemes, if the keys are known, and the communications stream has been completely intercepted, it becomes impossible to protect any system.

We believe that using a combination of the SKDP symmetric encryption scheme, and a quantum secure protocol like QSMP, used to introduce new entropy into the system periodically, that this hybrid scheme offers true long-term security. SKDP is a good candidate for any transaction-based protocol, where for example the embedded key can be stored on a debit or credit card. It is also a strong candidate for a private communications system, where a key is distributed on

pluggable memory-storage devices, and the device connects to a central communications hub. SKDP is also ideal for institutional VPN infrastructure, where private hub-and-spoke networks are created using a central office branching to remote locations. There are many different uses for this protocol, from remote control of machinery and infrastructure, a commodity trading system, to a flexible virtual private network configuration. Any application that requires a lightweight, high-security, scalable, post-quantum secure messaging system, is a prime candidate for SKDP.

SKDP is highly scalable, and can securely manage millions of devices through a single master key set. It derives branch keys from the master key, and device keys from the branch key. Any branch can connect to any client on any other branch, through a master server, by traversing a tree-like derivation structure. A branch key can be derived from the master key, and any leaf node on a branch, can have its embedded key derived by the branch server. Large networks can be scaled to regional institutions, those institutions can manage a collection of clients, with all clients on all branches being inter-accessible via a single root node that translates between them.

SKDP is a duplexed communications system. It uses a separate shared secret to key both the transmit and receive channels in a communications stream. Each server/host is responsible for generating the symmetric key that device transmits data on. Symmetric cipher keys are ephemeral, and unique keys are generated for each session. The system works in a client/server model, whereby a client requests a connection from the server to initiate the key exchange. The server authenticates and encrypts a key sent to the client, and the client encrypts and authenticates a key sent to the server. These keys are used to initialize a post-quantum secure symmetric cipher for each channel, which encrypts the communications stream. A strong emphasis has been placed on authentication with SKDP, with the entire key exchange using authentication to guarantee the exchange, and the symmetric stream cipher using KMAC authentication, with additional data parameters (AEAD) that authenticate the SKDP packet headers and payloads.

SKDP uses the NIST SHA3 Keccak family of cryptographic primitives to hash, permute, and generate pseudo-random, used in the key exchange portion of the SKDP protocol. Keccak is the most thoroughly studied, and cryptanalyzed family of cryptographic primitives in modern times. It underwent four years of intense analysis during the NIST SHA3 competition, and is widely considered to be one of the most secure and thoroughly studied family of cryptographic primitives in existence.

SKDP uses the authenticated symmetric stream cipher RCS, as the primary encryption service in the established encrypted tunnel. RCS is based on Rijndael, the cipher used in AES, with some important improvements. The transformation

function, the core function of a symmetric cipher, is the unaltered 256-bit wide Rijndael transformation function. The native differentially-weak key schedule, a cause of serious vulnerabilities in AES, has been replaced with cSHAKE, the Keccak extended output function, a cryptographically-strong key expansion function, which strongly mitigates many serious attacks that threaten AES. In addition to this, the number of transformation rounds has been increased, which in turn increases diffusion to the cipher-text output, placing algebraic based attacks out of reach. The number of transformation rounds has been increased from 14 used in AES-256, to 22 with a 256-bit key, and 30 rounds when using a 512-bit key, restoring the security margins to Rijndael that have been eroded away to dangerously small margins in AES.

RCS uses KMAC, the Keccak MAC function in a streaming AEAD mode, to authenticate the cipher-text in an encrypt-then-MAC configuration, that offers the strongest possible authentication. RCS and SKDP are capable of using 512-bit cipher keys, and we believe that at some future time, 512-bit keys will be necessary to ensure long-term security, and SKDP is ready, as a true 512-bit secure crypto-system.