

Elevating email to an enterprise-class database application solution

by George Sidman



When compared to the growth and maturity of other technologies, the Internet appears to be struggling with adolescence. It was born and nurtured on networks presumed to be private, exploded worldwide in the late nineties, and never had a chance to develop on sound engineering principals equal to its present importance and requirements.

Web applications and the cloud are now rapidly morphing towards true database solutions, while email is still stuck in the protocol layer of the OSI model (tinyurl.com/5bv8). Emails are sent out to traverse the Internet totally un-chaperoned and un-encrypted.

As fast and convenient as email is, it exacts a tremendous price as it forces us to protect ourselves against abuse, loss of privacy, financial fraud and fight other malicious behavior. Dollar losses to online fraud are already in the billions and show no signs of abating. We all pay for more than 90% of Internet traffic that is dangerous and unwanted, and no one is immune as this problem is the same for the individual and the corporation.

More and more observers and commentators suggest that the Internet is fundamentally broken - the security industry is no better than

90% effective against the one billion spam and fraud messages sent every day (tinyurl.com/5q8vys) - and we all wonder why the new protocols (IPv6, DNSSEC, IPsec, Domain Keys, etc.) have been so ineffective.

Current solutions get limited results

The DNS, critical to the routing mechanisms of the Internet, is also openly visible and also under constant attack. As a result, an email address presents itself as an open invitation to pillage and plunder. Malware blockers and filters are fighting a band-aid war they can never win.

Continuing to fight against a growing and increasingly innovative enemy will continue to be an expensive rear-guard action and a losing proposition.

Pursuing privacy and security through encrypting content or the transport wrapper (TLS, VPNs, etc.) has proven nearly worthless, as the exposure of the routing is the basis of the abuse. An email sent from a laptop over a VPN to a corporate server is safe until it must traverse the open Internet to reach a customer, partner or supplier (about 80% of commercial traffic.) It then leaves the DMZ to route in the clear, undoing the efficacy of the VPN. Legacy encryption solutions have failed to gain wide adoption, mostly because the technical challenges are beyond the average user. The market perceives that the true dangers of the Internet are really general abuse and fraud. Users are much less aware – and therefore concerned - that their email content might have value to someone other than the intended recipient.

Enter federal and state privacy laws

Federal and state agencies have begun enacting new privacy laws because there are

industries, such as healthcare and finance where the protection of personally identifiable information is critical (tinyurl.com/oyt9j).

The upturn of interest in electronic medical records includes compliance for HIPAA and SOX which is today driving an increased interest in private email. Web-based portals have emerged, under HTTPS, to deal with this requirement, but are only a partial solution. Adoption of portals is minimal, as doctors dislike clicking into a portal when their regular work habits include using email programs, such as Microsoft Outlook.

Portals are also only accessible to a small and select group of direct subscribers. But the larger issues are that only the transport layer is encrypted, leaving the central data storage in clear, usually in third-party hands, and not likely to pass a security audit. (See the EPIC complaint to the FTC regarding Google privacy claims - <http://tinyurl.com/yk8cnf5>).

Federal and state agencies have begun enacting new privacy laws because there are industries, such as healthcare and finance where the protection of personally identifiable information is critical.

This growing problem needs a new solution

There are hundreds of companies selling firewalls, VPNs, encryption solutions, malware blockers, and other security technologies and consulting services. The industry presumes that nothing can be done about the underlying problem, which is simply that the openly disclosed routing of email addresses, domain names and web addresses invites and supports abuse and fraud.

The next logical conclusion is that if the routing could be made private and the content hidden from view, the fraudsters would be thwarted and the abuse and fraud would be dramatically reduced, if not eliminated.

There is fundamentally nothing wrong with the basic engineering that underlies the Internet. Its protocols do a remarkable job of delivering connectivity and maintaining a high degree of integrity across billions of operations every

day. The problem is that the protocols (as described in the OSI model) are inadequate for the tasks at hand. They should be put to work in service of a broader software model. That model requires a true database application layer that wraps the protocols, providing an overlay of control facilities, bundling in encryption, key management, authentication, and certificates, as well as delivering on the new compliance requirements.

Email will most likely remain an adolescent technology until it acquires an application layer that lets it act like a true enterprise-class solution.

Ease-of-use is the holy grail of successful software. This has been achieved in a number of industries and solutions such as financial accounting software, CRM programs, online shopping and other day-to-day solutions which achieve that through the application layer.

It seems logical that we could achieve a new and much more powerful email capability simply by adding an application layer to the email protocols. The complexities of the components can be easily managed with database and applications code, removing the end-user from the technical challenges and masking the operations and content from prying eyes through private routing mechanisms and end-to-end encryption.

The dangers of the DNS

No standard email could route without the DNS and all web activity needs the DNS to translate names to IP addresses. Many view the DNS as sacrosanct, and so deeply imbed-

ded in Internet operations that even questioning its use is heresy. However, this is one Emperor who is indeed wearing no clothes. All DNS operations - from address lookups and resolver activity, to the Whois (beloved by fraudsters), and on to the many domain registrars (whose focus is revenues), across the (politically embattled) ICANN TLD – every aspect of the DNS is exposed on the open Internet.

That blatant visibility is the root cause of almost all malware - it fuels all fraud and cyber attacks and is the primary reason that no individual, enterprise or government is safe on the Internet.

**Nobody is going to fix the security flaws of the Internet.
It is here to stay as it is.**

Towards a new privacy model

Nobody is going to fix the security flaws of the Internet. It is here to stay as it is. Even with all its warts and problems it has driven new levels of information speed and freedom that the world has never before seen. We can use the Internet as it is, employing industry standards and open source code to create and deliver new levels of privacy, security and legal compliance.

The DNS can be left as it is. We don't necessarily need to use the name conversion facility to find a server. (In the private email space server IP addresses are few and easily managed – not by end-users, but safe within the application layer.) If we drop the DNS, we can then modify the email address so that it thwarts malware, simply because an email address without a TLD won't route publicly. We can render the addresses invisible through packet header encryption, along with the subject heading and other clues that might otherwise attract the wrong crowd.

These concepts are the first glimmers of privacy. By exploring this direction we could well create a new model that takes email away from its "Wild West" reputation and empowers it as a robust, safe and private means of communicating.

The components of the new architecture

To build a standard database application solution, we need central servers and a connectivity model, cloud or otherwise, that achieves layers of managed services. Today, the standard email server is James from Apache (james.apache.org). It is a protocol layer utility that utilizes flat files to route email to and from senders and recipients. The problem is that James and its cousins, SMTP/POP3/iMAP and others, all operate "in clear". They will accommodate encrypted content but the header must remain visible for routing under the DNS. Across all these protocols there is simply not enough information to deal with the additional needs now emerging for end-to-end privacy, compliance and reporting. Components that could be added to create a new architecture include:

- Adding a database to manage subscribers, policies, transaction logging, etc., James could take on a new life. Suddenly a whole new range of information is available to manage encryption, keys, authentication, user account services, etc.
- Adding a central key store within the database, PKI becomes easy to manage. Key distribution gets automated within the application layer and is no longer a burden to end-users.

- Encryption processes for both the content and the transport layer get handled within the application layer, eliminating the need for end-user involvement and much of the potential for errors. This enables the accuracy, efficiency and safety of large machine generated keys.
- Adding some dedicated server code we can resolve encrypted traffic before it gets to James and also manage connectivity to find addresses of remote services.
- Authentication and certificates also get embedded into the application layer, becoming more durable and reliable, free of user involvement and totally controlled by system administrators.
- Then, to extend the application layer to the desktop, we need a small piece of code that is downloaded and easily installed on desktops, laptops and other devices. It handles end-user side encryption and decryption, and other housekeeping functions through tight integration with the server-side code. By self installing, it handshakes standard email clients through standard ports and protocols. By utilizing standard email clients, end-user training is minimal and no substantial changes are required to existing business processes.

Under this new model, email becomes a complete ecosystem for privacy, security and compliance. It is a unified space into which all the scattered bits and pieces of our previous 'security toolkits' get integrated under a single application solution. This model operates in the OSI stack from the session layer up to the application layer. (VPNs operate from the session layer down.) As a result, such a solution is highly portable and independent of transport and connectivity.

Other major benefits

With a central database to log all transactions, the system can report on all email traffic on-

demand. For the first time, the life cycle of an email can be tracked, through replies and forwards, delivering on emerging compliance and eDiscovery requirements with ease. Reports can be rendered in various output formats for business intelligence purposes, and managers will have an enforcement mechanism to track end-user compliance to privacy policies.

With a network neutral application in place, plug-ins can be created for the various enterprise email services, such as Exchange, GroupWise, Domino and Citrix. This means that minimal disruption will occur in implementing a private email network and that private traffic will be easily managed alongside standard email services.

Freedom from malware is one of the major benefits of such a model. The standard Internet model of anonymity and non-accountability is inverted. In this new 'Gated Community' all users are known and fully accountable. A rogue subscriber can simply be shut down.

If the two ends, the client-side and the server-side, are indeed closely coupled through application code and encryption - and all routing is protected through non-DNS addressing and other controls are in place - then privacy is truly achieved. After all, do you have privacy if publicly visible routing exposes who you are, and you cannot control who sends you email?

The Internet has been mostly under the control of network engineers. Email would most likely benefit hugely if application and security software engineers took a stronger hand. Bringing the email protocols together in a database application is the next logical step and until that happens we be subject to the risks and dangers of abusive and costly email problems.

George Sidman is the Chairman and Chief Technology Architect at WebLOQ (www.webloq.com). His technology experience spans large-scale library and information automation, commercial ISP services, and Internet security and privacy technologies. He is also a licensed Architect, and is the former Chair of the Technology Council of the Silicon Valley World Internet Center in Palo Alto, California. He sits on the Boards of the Marina Technology Cluster in Marina, California, and other technology companies.