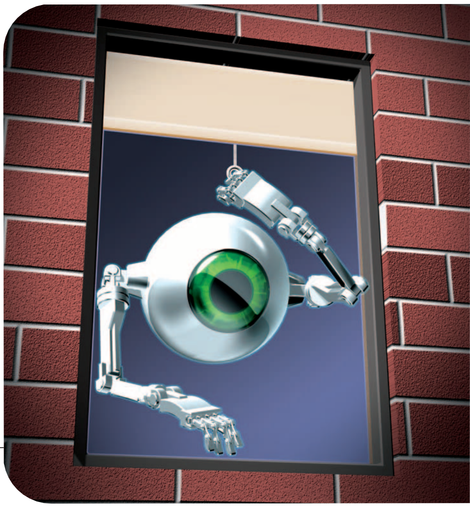
Challenges in Securing Voice over IP

Although VoIP offers lower cost and greater flexibility, it can also introduce significant risks and vulnerabilities. This article explains the challenges of VoIP security and outlines steps for helping to secure an organization's VoIP network.



Thomas J. Walsh and D. Richard Kuhn
National Institute of Standards and Technology

This article is about security, more specifically, about protecting one of your most precious assets—your privacy. We guard nothing more closely than our words. One of the most important decisions we make every day is what we will say and what we won't. But even then it's not only what we say, but also what someone else hears, and who that person is.

Voice over IP—the transmission of voice over traditional packet-switched IP networks—is one of the hottest trends in telecommunications. Although most computers can provide VoIP and many offer VoIP applications, the term "voice over IP" is typically associated with equipment that lets users dial telephone numbers and communicate with parties on the other end who have a VoIP system or a traditional analog telephone. (The sidebar, "Current voice-over-IP products," describes some of the products on the market today.)

As with any new technology, VoIP introduces both opportunities and problems. It offers lower cost and greater flexibility for an enterprise but presents significant security challenges. Security administrators might assume that because digitized voice travels in packets, they can simply plug VoIP components into their already secured networks and get a stable and secure voice network. Unfortunately, many of the tools used to safeguard today's computer networks—firewalls, network address translation (NAT), and encryption—don't work "as is" in a VoIP network. Although most VoIP components have counterparts in data networks, VoIP's performance demands mean you must supplement ordinary network software and hardware with special VoIP components.

Integrating a VoIP system into an already congested or overburdened network can be disastrous for a company's technology infrastructure. Anyone attempting to construct a VoIP network should therefore first study the procedure in great detail. To this end, we've outlined some of the challenges of introducing appropriate security measures for VoIP in an enterprise.

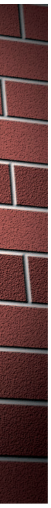
Supporting protocols

Most current VoIP systems use one of two protocols. H.323[1] is the International Telecommunication Union (ITU) specification for audio and video communication across packetized networks. It acts as a wrapper for a suite of ITU media control recommendations. Each protocol has a specific role in the call-setup process.

An H.323 network consists of several end points (terminals) that are normally bound to a specific address and gateway, and possibly a gatekeeper, multipoint control unit, and back-end service. The gateway serves as a bridge between the H.323 network and the outside world of (possibly) non-H.323 devices, including Session Initiation Protocol (SIP) networks and traditional public switched telephone networks (PSTNs).

VoIP systems also use SIP,[2] the IETF-specified protocol for initiating a two-way communication session. SIP was designed to be simpler than H.323, but it has become increasingly complex as it has evolved. Being text based, SIP avoids some Abstract Syntax Notation number One (ASN.1) parsing issues[3] that occur with the H.323 protocol suite if S/MIME isn't used.

SIP is an application-level protocol—that is, it's decoupled from the protocol layer that it's transported across. The Transmission Control Protocol (TCP), User Datagram Protocol (UDP), or Stream Control Transmission Protocol



Current voice-over-IP products

Demand for VoIP services has resulted in a broad array of products, including

- *Telephone handsets*. These products are usually more than a simple handset with dial pad. Some of these units have a base station design, providing the convenience of a conventional cordless phone.
- *Conferencing units*. These units provide the same type of service as conventional conference calling phone systems, but because communication takes place over the Internet, users can coordinate traditional data communication services, such as a whiteboard displayed on computer monitors at both ends.
- *Mobile units.* Wireless VoIP units are becoming increasingly popular, especially because many organizations already have an installed base of 802.11 networking equipment. Wireless VoIP products present particularly acute security problems, given the well-known weaknesses of the 802.11b protocols.
- *PC or softphone.* With a headset, software, and inexpensive connection service, any PC or workstation can serve as a VoIP unit, or softphone.

In addition to end-user devices, VoIP systems also include specialized components such as call managers and media/signaling gateways. Call managers set up calls, monitor call state, handle number translation, and provide basic telephony services. They also handle signaling functions that coordinate with media gateways—the interface between the VoIP network and the public switched telephone network (PSTN). Depending on the system, designers can implement the gateway functions as a board or dedicated appliance or provide them through a distributed system.

(SCTP), a newer specification designed to transport signaling protocols, can carry it. You can use UDP to decrease overhead and increase speed and efficiency, or you can use TCP if you incorporate Secure Sockets Layer/Transport Layer Security (SSL/TLS) for security services. Unlike H.323, SIP uses only one port in the call-setup process.

The architecture of a SIP network also differs from the H.323 structure. A SIP network consists of end points, a proxy or redirect server, a location server, and a registrar. In the SIP model, a user isn't bound to a specific host. Instead, users initially report their locations to a registrar, which can be integrated into a proxy or redirect server.

VoIP vs. data network security

To understand why security for VoIP differs from data network security, we need to look at the unique constraints of transmitting voice over a packet network, as well as the characteristics shared by VoIP and data networks.

Packet networks depend on many configurable parameters: IP and MAC (physical) addresses of voice terminals and addresses of routers and firewalls. VoIP networks add specialized software, such as call managers, to place and route calls. Many network parameters are established dynamically each time a network component is restarted or when a VoIP telephone is restarted or added to the network. Because so many nodes in a VoIP network have dynamically configurable parameters, intruders have as wide an array of potentially vulnerable points to attack as they have with data networks. But VoIP systems have much stricter performance constraints than data networks, with significant implications for security.

Quality-of-service issues

Quality of service (QoS) is fundamental to a VoIP network's operation. A VoIP application is much more sensitive to delays than its traditional data counterparts. When downloading a file, a few seconds' slowdown is negligible. In contrast, a mere 150-millisecond delay can turn a crisp VoIP call into a garbled, unintelligible mess.[4] In the VoIP vernacular, this is the *latency problem*.

Latency turns traditional security measures into double-edged swords for VoIP. Tools such as encryption and firewall protection can help secure the network, but they also introduce significant delay. Latency isn't just a QoS issue, but also a security issue because it increases the system's susceptibility to denial-of-service attacks. To succeed in a VoIP network, a DoS attack need not completely shut down the system, but only delay voice packets for a fraction of a second. The necessary impediment is even less when latency-producing security devices are slowing down traffic.

Another QoS issue, *jitter,* refers to nonuniform delays that can cause packets to arrive and be processed out of sequence. The Real-Time Transport Protocol (RTP), which is used to transport voice media, is based on UDP, so packets received out of order can't be reassembled at the transport level, but must be reordered at the application level, introducing significant overhead. Even when packets arrive in order, high jitter causes them to arrive at their destination in spurts. To control jitter, network designers can use buffers and implement QoS-supporting network elements (especially routers) that let VoIP packets "play through" when larger data packets are scheduled ahead of them. The buffers can use one of several strategies to determine when to release voice data, including several schemes that adapt the playout time during a conversation.[5]

QoS also encompasses *packet loss*. In addition to the traditional packet loss issues associated with data networks, even VoIP packets that reach their destinations can be rendered useless by latency and jitter. Compounding the packet loss problem is VoIP's reliance on RTP, which doesn't guarantee packet delivery.

The good news is that VoIP packets are small, containing a payload of only 10–50 bytes, or approximately 12.5–62.5 ms, with most implementations at the shorter end of the range. The loss of such a minuscule amount of speech is indiscernible, or at least unworthy of complaint, by a human VoIP user.

The bad news is that these packets are rarely lost in isolation. Most causes of packet loss affect all packets being delivered around the same time. So, although losing one packet is fairly inconsequential, probabilistically it means the loss of several packets, which severely degrades a VoIP network's QoS. Packet losses as low as 1 percent can make a call unintelligible, depending on the compression scheme used. A 5-percent loss is catastrophic, no matter how good the codec. This sensitivity increases a DoS attack's effectiveness. To be successful, a DoS attack needs to flood or disrupt the network only enough to stop 5 percent of packets from being delivered on time.

Thus, an enterprise's hardware should support QoS to deliver VoIP traffic at high speed and with preference over less urgent data traffic. An enterprise can use routers that forward packets based on type-of-service (ToS) bits, for example, or provide a separate queue for VoIP traffic. Anton Kos and colleagues significantly reduced jitter and latency using priority-based network elements.[5] Call Admission Control can help minimize packet loss by detecting network saturation and preventing VoIP packets from embarking on journeys they can't complete. Victoria Fineberg covers CAC and other network-specific QoS issues in more depth.[6]

Infrastructure issues

With conventional telephones, eavesdropping requires tapping a line or penetrating a switch. Attempting such physical access increases the intruder's risk of discovery. Conventional PBXs typically use proprietary protocols and specialized software and have fewer points of access than VoIP systems. With VoIP, opportunities for eavesdroppers are multiplied. VoIP units share physical network connections with the data network, and in many cases VoIP and data are on the same logical portion of the network. Protocols are standardized, and tools to monitor and control packet networks are widely available. Attaching a packet sniffer to the VoIP network segment makes it easy to intercept voice traffic.

Good quality open source packages are available for such monitoring, including both SIP and H.323 plug-ins for packet sniffers such as the popular Ethereal analyzer (www.ethereal.com). Voice over misconfigured Internet telephones (http://vomit.xtdnet.nl), a publicly available utility with an unfortunate acronym—Vomit—converts standard tcpdump (http://sourceforge.net/projects/tcpdump) format files into .wav files that any computer can play. Tcpdump is a standard Linux utility and is freely available for Windows systems, making VoIP eavesdropping accessible to anyone with a PC and an Internet connection.

VoIP is similar to consumer software in how it's distributed and installed. Like printer or digital camera drivers, software loads for VoIP phones are typically available on the Internet. This arrangement makes it easy for customers to download updates but it also raises some security risks. A technically skilled person can download software loads, disassemble them, make malicious modifications, and then use the hacked version to attack phones on a VoIP network. Because software integrity often relies on a cyclic redundancy check rather than a cryptographically strong hash code, an attacker can compromise the network by building an executable image that meets the CRC.

Investigations by the University of Tulsa and the US National Institute of Standards and Technology found that attackers can use most of the familiar attacks on TCP/IP networks to insert a hacked binary into the system.[7] For example, when reset (possibly due to network failure), VoIP phones can receive configuration files and software loads through the Trivial File Transfer Protocol (TFTP). An attacker with physical access to any part of the network segment can use the Address Resolution Protocol (ARP) cache-poisoning techniques (changing the MAC address associated with a particular IP address) to substitute a rogue server for the correct one, causing phones to download the hacked software. An even easier attack is to set up a rogue server with modified configuration files containing the IP addresses of call managers. Victims' calls are then routed through the attacker's call manager, providing eavesdropping and traffic analysis opportunities.

A related vulnerability arises from the availability of network information on IP phones. Each phone stores the TFTP server's location, and many VoIP phones use minimal access control, such as a constant digit string. An attacker with one-time physical access to the phone can replace the TFTP server's location, causing the phone to download from a compromised or attacker-created server.[8] Similarly, a rogue Dynamic Host Configuration Protocol server can exploit race conditions, because phones often use DHCP on booting to obtain their IP address. Another documented vulnerability in VoIP systems is the ability to spoof the caller ID number using three-way calling features.[9]

Like other types of software, VoIP systems have vulnerabilities due to buffer overflows and improper packet header handling. Exploitable software flaws typically result in two types of vulnerabilities: DoS and disclosure of critical system parameters.

Researchers at Oulu and Columbia Universities analyzed numerous SIP implementations and found that passing specially constructed or malformed packets can cause the software to fail, resulting in vulnerabilities that include DoS, unauthorized access, and remote code execution.[10–12] In some cases, the system crash produces a memory dump containing IP addresses of critical system nodes, passwords, or other security-relevant information. Crashing a VoIP server can also cause a restart that restores default passwords or falls prey to a rogue server attack. In addition, buffer overflows can be used to introduce malicious code in VoIP software.

Security trade-offs

Trade-offs between convenience and security are routine in software, and VoIP is no exception. Most, if not all, VoIP components use integrated Web servers for configuration. Web interfaces can be attractive, easy to use, and inexpensive to produce because of the wide availability of good development tools. Unfortunately, most Web development tools focus on features and ease of use, with less attention paid to the security of the applications they help produce. Some VoIP device Web applications have weak or no access control, script vulnerabilities, and inadequate parameter validation, resulting in privacy and DoS vulnerabilities.[8] Some VoIP phone Web servers use only HTTP basic authentication,[13] meaning servers send authentication information without encryption, letting anyone with network access obtain valid user IDs and passwords. As VoIP gains popularity, we'll inevitably see more administrative Web applications with exploitable errors.

Need for new technologies

Firewalls are a security staple in today's IP networks. Whether protecting a local or wide-area network, encapsulating a demilitarized zone (DMZ)—that is, a computer or small network acting as a buffer between a private network and the Internet—or just protecting a single computer, a firewall is usually the first line of defense. Firewalls block traffic deemed malicious or potentially risky. A set of rules—such as "block all FTP traffic (port 21)" or "allow all http traffic (port 80)"—programmed into the firewall by the network administrator determines acceptable traffic. Firewalls also provide a central location for deploying security policies. They're the ultimate network traffic bottleneck because traffic can't enter or exit the LAN without passing them.

In a VoIP network, firewalls simplify security management by consolidating security measures at the firewall gateway instead of requiring the end points to maintain up-to-date security policies. Unfortunately, introducing firewalls to a VoIP network complicates several aspects of VoIP, most notably dynamic port trafficking and call-setup procedures. Several commercial solutions can alleviate this problem. For example, Application Level Gateways (ALGs) make firewalls "VoIP aware" and Midcom Controls[14] let VoIP packets traverse the firewall by letting the firewall receive instructions from an application-aware agent—that is, because they understand the VoIP protocol data carried as payload in an ordinary packet, they can perform stateful filtering of call packets. Implementing a VoIP system on a legacy network without such devices is generally not feasible.

NAT is a powerful tool for hiding internal network addresses and letting several end points within a LAN use the same (external) IP address. NATs change outgoing IP headers from private LAN addresses to the router's global IP address, letting several computers simultaneously share the address. In addition, machines that don't need to access the Internet can be assigned local intranet addresses, avoiding conflicts and keeping IP addresses open for machines that need them. NATs also provide an added layer of security for LANs, making internal IP addresses inaccessible on the public Internet. Thus, attacks against the network must focus on the NAT router itself. Like firewalls, this increases security because you need protect only a single access point. Moreover, the router is usually far more secure than a PC directly connected to the Internet.

NATs' benefits come at a price, however. For one thing, they make calls into the network very complex, as in an office phone network in which many extensions share the same external phone number. Other issues are associated with media transmission across the NAT, including an incompatibility with IPsec. To resolve these problems, you can use NAT ALGs or force address translation away from the NAT using serial tunneling (STUN), Traversal Using Relay NAT (TURN), or midcom, although you must handle the incompatibility with IPsec at the protocol level.

Although firewalls, gateways, and other such devices can help prevent intruders from compromising a network, they're no defense against internal hackers and don't protect voice data as it crosses the Internet. Protecting the data itself requires a layer of defense at the protocol level. In VoIP, as in data networks, you can encrypt the packets at the IP level using IPsec, making them unintelligible to all but the intended recipient. The IPsec suite of security protocols and encryption algorithms is the standard for securing packets against unauthorized viewers over data networks and will be supported by the protocol stack in IPv6. It therefore seems logical to extend IPsec to VoIP.

Introducing firewalls to a VoIP network complicates several aspects of VoIP, most notably dynamic port trafficking and call-setup procedures.

However, because routers, proxies, and other components must read the VoIP packets, packets are often encrypted at a network's gateways, rather than its end points. This scheme also lets the end points stay computationally simple and promotes scalability because you can overlay new encryption algorithms on the network without upgrading the end points. Unfortunately, several factors, including packet size expansion, ciphering latency, and a lack of QoS urgency in the cryptographic engine can cause an excessive amount of latency in VoIP packet delivery, leading to degraded voice quality.

The encryption process can be detrimental to QoS, making cryptodevices severe bottlenecks.

The encryption process can be detrimental to QoS, making cryptodevices severe bottlenecks in a VoIP network. Encryption latency is introduced at two points. First, encryption and decryption take a nontrivial amount of time. VoIP's multitude of small packets exacerbates the encryption slowdown because most of the time consumed comes as overhead for each packet. One way to avoid this slowdown is to apply computationally simple encryption algorithms to the voice data before packetization.[15] Although this improves throughput, the proprietary encryption algorithms used (fast Fourier-based encryption, chaos-bit encryption, and so on) aren't considered as secure as the Advanced Encryption Standard,[16] which is included in many IPsec implementations. AES's combination of speed and security should handle the demanding needs of VoIP at both ends.

Recent studies indicate that the greatest contributor to the encryption bottleneck occurs at the cryptoengine scheduler, which often delays VoIP packets as it processes larger data packets.[17] This problem stems from the fact that cryptoschedulers are usually first-in first-out (FIFO) queues, inadequate for supporting QoS requirements. If VoIP packets arrive at the encryption point when the queue already contains data packets, there's no way they can usurp the less time-urgent traffic. Some hardware manufacturers have proposed (and at least one has implemented) solutions for this, including QoS reordering of traffic just before it reaches the cryptoengine.[18] But this solution assumes that the cryptoengine's output is fast enough to avoid saturating the queue. Ideally, you'd want the cryptoengine to dynamically sort incoming traffic and force data traffic to wait for it to finish processing the VoIP packets, even if these packets arrive later. However, this solution adds considerable overhead to a process most implementers like to keep as light as possible. Another option is to use hardware-implemented AES encryption, which can improve throughput significantly.

Past the cryptoengine stage, the system can perform further QoS scheduling on the encrypted packets, provided they were encrypted using ToS preservation, which copies the original ToS bits into the new IPsec header.

Virtual private network (VPN) tunneling of VoIP has also become popular recently, but the congestion and bottlenecks associated with encryption suggest that it might not always be scalable. Although researchers are making great strides in this area, the hardware and software necessary to ensure call quality for encrypted voice traffic might not be economically or architecturally viable for all enterprises considering the move to VoIP.

Thus far, we've painted a fairly bleak picture of VoIP security. We have no easy "one size fits all" solution to the issues we've discussed in this article. Decisions to use VPNs instead of ALG-like solutions or SIP instead of H.323 must depend on the specific nature of both the current network and the VoIP network to be. The technical problems are solvable, however, and establishing a secure VoIP implementation is well worth the difficulty.

To implement VoIP securely today, start with the following general guidelines, recognizing that practical considerations might require adjusting them:

- Put voice and data on logically separate networks. You should use different subnets with separate RFC 1918 address blocks for voice and data traffic and separate DHCP servers to ease the incorporation of intrusion-detection and VoIP firewall protection.
- At the voice gateway, which interfaces with the PSTN, disallow H.323, SIP, or Media Gateway Control Protocol (MGCP) connections from the data network. As with any other critical network management component, use strong authentication and access control on the voice gateway system.
- Choose a mechanism to allow VoIP traffic through firewalls. Various protocol dependent and independent solutions exist, including ALGs for VoIP protocols and session border controllers. Stateful packet filters can track a connection's state, denying packets that aren't part of a properly originated call.
- Use IPsec or Secure Socket Shell (SSH) for all remote management and auditing access. If practical, avoid using remote management at all and do IP PBX access from a physically secure system.
- Use IPsec tunneling when available instead of IPsec transport because tunneling masks the source and destination IP addresses, securing communications against rudimentary traffic analysis (that is, determining who's making the calls).
- If performance is a problem, use encryption at the router or other gateway to allow IPsec tunneling. Because some VoIP end points aren't computationally powerful enough to perform encryption, placing this

burden at a central point ensures the encryption of all VoIP traffic emanating from the enterprise network. Newer IP phones provide AES encryption at reasonable cost.

- Look for IP phones that can load digitally (cryptographically) signed images to guarantee the integrity of the software loaded onto the IP phone.
- Avoid softphone systems (see the sidebar) when security or privacy is a concern. In addition to violating the separation of voice and data, PC-based VoIP applications are vulnerable to the worms and viruses that are all too common on PCs.
- Consider methods to harden VoIP platforms based on common operating systems such as Windows or Linux. Try, for example, disabling unnecessary services or using host-based intrusion detection methods.
- Be especially diligent about maintaining patches and current versions of VoIP software.
- Evaluate costs for additional power backup systems that might be required to ensure continued operation during power outages.
- Give special consideration to E-911 emergency services communications, because E-911 automatic location service is not always available with VoIP.

VoIP can be done securely, but the path isn't smooth. It will likely be several years before standards issues are settled and VoIP systems become mainstream. Until then, organizations must proceed cautiously and not assume that VoIP components are just more peripherals for the local network. Above all, it's important to keep in mind VoIP's unique requirements, acquiring the right hardware and software to meet the challenges of VoIP security. □

Acknowledgments

We thank Tim Grance for supporting this work. We are especially grateful to Mike Stauffer and Steffen Fries, who helped tremendously in improving our understanding of many aspects of VoIP operation.

References

1. *Recommendation ITU-T H.323, Packet-Based Multimedia Communications Systems*, Int'l Telecomm. Union, 1998.
2. J. Rosenberg et al., *Session Initiation Protocol*, IETF RFC 3261; www.ietf.org/rfc/rfc3261.txt.
3. Vulnerability Note VU#749342, "Multiple Vulnerabilities in H.323 Implementations," US Computer Emergency Readiness Team, www.kb.cert.org/vuls/id/749342.
4. *Recommendation ITU-T G.114, One Way Transmission Time*, Int'l Telecomm. Union, 1988.
5. A. Kos, B. Klepec, and S. Tomazic, "Techniques for Performance Improvement of VoIP Applications," *IEEE Mediterranean Electrotechnical Conf.*, IEEE Press, 2002, pp. 250–254.
6. V. Fineberg, "Building a QoS-Enabled IP Network: A Practical Architecture for Implementing End-to-End QoS in an IP Network," *IEEE Comm.*, Jan. 2002, pp. 122–130.
7. D.R. Kuhn, T.J. Walsh, and S. Fries, *Security Considerations for Voice Over IP Systems*, SP 800-58, US Nat'l Inst. Standards and Technology, Dec. 2003.
8. Bugtraq, "Multiple Vulnerabilities in Cisco VoIP Phones," May 2002, http://cert.uni-stuttgart.de/archive/bugtraq/2002/05/msg00219.html.
9. N. Wosnack, "Bugtraq: A Vonage VoIP 3-Way Call CID Spoofing Vulnerability," Aug. 2003, http://seclists.org/lists/bugtraq/2003/Aug/0274.html.
10. *PROTOS: Security Testing of Protocol Implementations*, tech. report, Univ. of Oulu, Jan. 2005; www.ee.oulu.fi/research/ouspg/protos/index.html.
11. C. Wieser, M. Laakso, and H. Schulzrinne, *Security Testing of SIP Implementations*, tech. report CUCS-24-03, Dept. of Computer Science, Columbia Univ., 2003.
12. CERT Advisory CA-2003-06, "Multiple Vulnerabilities in Implementations of the Session Initiation Protocol (SIP)," www.cert.org/advisories/CA-2003-06.html.
13. O. Arkin, "More Vulnerabilities with Pintgel xpressa SIP-based IP Phones," Sys-Security Group, 2002, www.sys-security.com/archive/advisories/More_Vulnerabilities_with_Pingtel_xpressa_Phones.pdf.
14. "MIDCOM: Middlebox Comm. Working Group," www.ietf.org/html.charters/midcom-charter.html.
15. J.-I. Guo, J.-C. Yen, and H.-F. Pai, "New Voice over Internet Protocol Technique with Hierarchical Data Security Protection," *IEE Proc. Vision, Image, & Signal Processing*, vol. 149, no. 4, Aug. 2002, pp. 237–243.
16. Nat'l Inst. of Standards and Technology, *Advanced Encryption Standard*, Federal Information Processing Standard 197, Nov. 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.
17. R. Barbieri, D. Bruschi, and E. Rosti, "Voice over IPsec: Analysis and Solutions," *Proc. 18th Ann. Computer Security Applications Conf.*, IEEE CS Press, 2002, pp. 261–270.
18. Cisco, "Reference Guide to Implementing Crypto and QoS," document ID 18667, Oct. 2002, www.cisco.com/warp/public/105/crypto_qos.html.

***Thomas J. Walsh** is a PhD candidate at Rutgers University and former research fellow at the US National Institute of Standards and Technology. His research interests include machine learning, cognitive similarity models, and computer security. He is a member of the Rutgers Laboratory for Real Life Reinforcement Learning (RL3). Walsh has a BS in computer science from the University of Maryland, Baltimore County. Contact him at thomaswa@cs.rutgers.edu.*

***D. Richard Kuhn** is a computer scientist in the Computer Security Division of the US National Institute of Standards and Technology. His primary technical interests are in information security and software assurance. Kuhn received an MS in computer science from the University of Maryland at College Park and an MBA from the College of William and Mary. He is coauthor of the book* Role Based Access Control *(Artech House, 2003) and a senior member of the IEEE and the IEEE Computer Society. Contact him at kuhn@nist.gov.*