



# IEEE Security and Privacy Symposium in the Year 2003

Terry Benzel   
Hilarie Orman 

In noting the 20th anniversary of the establishment of *IEEE Security & Privacy Magazine*, we give a retrospective view of the research papers from the Security and Privacy Symposium in 2003. These papers represent the context of security concerns and solutions of that era. In some cases they illustrate problems that were solved; in other cases they foreshadow the dominant themes of today.

Stepping back 20 years, we find ourselves in a primitive world. There are no cryptocurrencies, few websites use Java, cellphones don't run apps, there is no Internet of Things because there are no "Things," the dot-com bubble is a recent and painful cautionary tale, Windows 2000 is ubiquitous, elliptic curve techniques for cryptography are new and barely understood, and quantum computing is science fiction. Where were security and privacy in this era that was on the cusp of massive improvements in processing power and communications? The Security and Privacy Symposium that year published 19 papers that showed foresight and practical relevance in varying degrees.

Two papers on denial of service provided important mitigation approaches for a relatively new research area. Both papers provided new techniques. In "Defending Against Denial-of-Service Attacks With Puzzle Auctions," Wang and Reiter<sup>1</sup> proposed the use of client puzzles with the addition of auctions and demonstrated this in Linux. This work is notable because, while puzzle solving never played a role in combatting distributed-denial-of-service (DDoS), it is very relevant to proof-of-work blockchains: it is necessary to solve a puzzle to add a block to the chain or to mine for cryptocurrency. The paper by Wang and Reiter describes adaptive puzzles, which are precisely what blockchains

do. The second paper, "Pi: A Path Identification Mechanism to Defend Against DDoS Attacks" by Yaar et al.,<sup>2</sup> proposed the use of packet marking to defend against DDoS. Experimental results demonstrated the effectiveness of the approach even when only 50% of the routers in a topology do not participate in the marking. Both papers have been extensively referenced, with more than 600 citations per Google Scholar.

There were three papers on trust management and protection of policies. In "A Unified Scheme for Resource Protection in Automated Trust Negotiation," Yu and Winslett<sup>3</sup> presented a unified scheme to model the protection of resources, including policies, in trust negotiation. Modeling policies as first-class resources allowed fine-grained control over policy disclosure and clearly distinguished between policy disclosure and policy satisfaction, which gives users more flexibility in expressing their authorization requirements. Then, in "Beyond Proof-of-Compliance: Safety and Availability Analysis in Trust Management," Li et al.<sup>4</sup> studied security properties, such as safety and availability, for a family of trust management languages, devising algorithms for deciding the possible consequences of certain policy changes. The importance of this contribution is that it developed an approach to security analysis focusing on availability rather than strict safety. Thus, the paper provides proof for a decidable system in contrast to the classical Harrison, Ruzzo, Ullman access matrix model for which safety queries are undecidable.

Digital Object Identifier 10.1109/MSEC.2023.3237103  
Date of current version: 15 March 2023

In a third related paper on security policies, “Hardening Functions for Large-Scale Distributed Computations,” Szajda et al.<sup>5</sup> developed an approach to specifying and enforcing system-wide security policies in distributed systems with mutual distrust. This paper described a way to enforce policies for data confidentiality and integrity in a distributed system. The approach relies on compiler-generated secure run-time protocols for communication among replicated code partitions.

Moving from policies, the next several papers addressed distributed systems and Internet-connected systems. In “Using Replication and Partitioning to Build Secure Distributed Systems,” Zheng et al.<sup>6</sup> developed an approach to hardening computations running across a large number of shared personal computers. The application of concern was the use of spare computing cycles (to create computing power akin to supercomputers) and methods for guaranteeing the integrity and accounting of those computations. The algorithms include seeding certain tasks with precomputed data and a strategy of sharing the computation of  $N$  tasks among  $K > N$  nodes.

Another paper along similar lines and related to the DDoS problem, “Garbage Collector Memory Accounting in Language-Based Systems” by Price et al.,<sup>7</sup> addressed techniques to measure and control resource usage, particularly to account for and regulate heap memory usage on a per-task basis. The authors described how the lack of protection around this mechanism in compilers could be exploited by a misbehaving task, which might allocate and hold enough memory to cause a denial-of-service attack, crashing or slowing down other tasks. The paper presented a method for modifying the garbage collector, present in modern (at the time) language runtime systems, to measure the amount of reachable memory for each task. The author’s prototype demonstrates a negligible performance overhead while providing enough information for the expression of rich policies to express the limits on a task’s memory usage.

Interestingly, there were few papers addressing operating systems at the 2003 Symposium. In “Vulnerabilities in Synchronous IPC Designs,” Shapiro<sup>8</sup> analyzed, previous to 2003, interprocess communication (IPC) designs in five microkernel architectures (EROS, Mach, L4, Flask, and Pebble). He proposed a new design for EROS that ensures IPC asymmetric trust, reproducibility, and support for dynamic payload lengths. This is achieved through authentication of the code executed by an application independent of its user and the ability (via confinement) to protect a trusted program from tampering by its user. The author stated that, to their knowledge, “no previous papers have been published that expose in depth or adequately address the interprocess denial-of-service vulnerabilities that are implicit in synchronous IPC.”

In an “attack” paper, “Using Memory Errors to Attack a Virtual Machine,” Govindavajhala and Appel<sup>9</sup> described using memory errors to attack a virtual machine. The authors presented an experimental study showing that soft memory errors can lead to security vulnerabilities in systems that rely on the type checking of untrusted programs as a protection mechanism. The authors suggest that these sorts of attacks are particularly relevant against smart cards or tamper-resistant computers. The paper describes a demonstration of the attack by sending a Java program that is designed so memory errors in its address space will allow it to take control of the Java virtual machine. The authors argue that the attack technique is applicable against other language-based security schemes. Note that, for the attack to occur, there must be memory errors. The authors described multiple possible sources of the introduction of memory errors. The authors conclude, “The best defense is the use of hardware error-detection and correction . . . with software logging of errors and appropriate response to unusual patterns of errors.”

In “Specifying and Verifying Hardware for Tamper-Resistant Software,” Lie et al.<sup>10</sup> specified a hardware architecture that supports tamper-resistant software. High-integrity software for critical applications needs to run on a processor that can protect the software and the data it manipulates. A hardware architecture that can provide this protection for the off-chip memory, cache, and registers was designed in 2000. In the model used by this paper, software is protected by an “execute-only” architecture, eExecute Only Memory (XOM). When data move off chip, they are tagged and encrypted. When they move back, they are decrypted, and the tag is retained. The processor enforces a security policy that compartmentalizes data by tag. Using a model checker (Mur-phi), the authors were able to find and fix a flaw that allowed policy violation, and they were also able to show that one of the security checks was not necessary. They also found ways to help ensure liveness. It is interesting to note that this work preceded today’s mobile devices running mutually distrusting apps. The XOM architecture is used in some high-integrity devices today. Choosing to use formal methods in this architecture seems prescient for its time.

Onion routing was seen as a pathway to anonymous communication, and its evolution into Tor was on the horizon. However, would this communication truly be anonymous if observers were trying to correlate traffic patterns? Today, there is a practical need for law enforcement to achieve this correlation to trace cryptocurrency transactions for contraband sold through the “dark web,” but, in 2003, anonymous networks were seen as a way to protect private communication from government surveillance.

David Chaum proposed anonymous communication networks in 1981, and, by 2003, the idea had become instantiated in the Naval Research Laboratory's onion routing. However, anonymity's adversary was the passive observer who could see enough of the communication to deduce which endpoints were engaged in point-to-point communication. What wasn't clear was how much observation was needed to do this deduction. In "Probabilistic Treatment of MIXes to Hamper Traffic Analysis," Agrawal et al.<sup>11</sup> derived some answers for anonymous networks by defining the "disclosure attack," which turns out to be an NP-complete problem. This posed limits on the sizes of their simulations, but they were still able to derive interesting results. Nonetheless, they noted that the attack was suboptimal and still allowed some probabilistic information to be derived. In the same symposium session, the paper "Defending Anonymous Communications Against Passive Logging Attacks" by Wright et al.<sup>12</sup> used real web access logs to determine if modifications to path selection could defeat adversaries who had some ability to observe traffic. They widened the network model to include dynamic node sets with varying path selection. Finally, in "Mixminion: Design of a Type III Anonymous Remailer Protocol," Danezis et al.<sup>13</sup> took a look back at designs for hiding e-mail communication paths. They noted that previous designs had fixable privacy errors, and they put forth a new design. Although remailers did not catch on, Dingedline was on the verge of creating the much more useful anonymous network, Tor.

Intrusion detection might have seemed like a solvable problem 20 years back, but it is a never-ending battle. Two papers defined defense mechanisms that were incremental improvements, one in network traffic monitoring and one in runtime local monitoring. "Active Mapping: Resisting NIDS Evasion Without Altering Traffic" by Shankar and Paxson<sup>14</sup> focused on TCP/IP implementations that had observable differences. Their method allowed a firewall to assign a TCP/IP classification to each host on a local network and to note any anomalies concerning its implementation policy.

Another paper showed that, by using local runtime artifacts, a system monitor could detect unusual calling returns. "Anomaly Detection Using Call Stack

Information" by Feng et al.<sup>15</sup> recommended building a table of runtime information about call sequences so that suspicious sequences can be distinguished from legitimate ones. This is complicated by system calls, such as signal handlers, that do not return to their caller.

The "virtual path" method builds a model of call sequences during training and can detect a wider class of anomalies than competing methods of the time. In "Intransitive Noninterference for Cryptographic Purposes," Backes and Pfitzmann<sup>16</sup> described a formal model of mediated communication channels that included cryptographic authentication. The purpose of this was to enable model checkers to determine if private information was leaked. By including the authentication data in

the formal model of the policy, the authors were able to check the correctness of some common uses of firewalls, message guards, and downgraders.

Identity-based encryption based on elliptic curve pairing cryptography gained some traction for e-mail identities, but it had an even more interesting use for

selective disclosure of attributes. "Secret Handshakes From Pairing-Based Key Agreements" by Balfanz et al.<sup>17</sup> was a seminal paper that showed that end users could securely control their certified personal information.

Software-based cryptography had become commonplace during the 90s as computers became much faster in carrying out arithmetic operations. However, there was a growing appetite for implementations on small, low-power processors for electronic gadgets and smartcards. Two papers discussed methods for extending cryptography into these niche areas. The University of California, Berkeley had developed some low-power chips with wireless communication for device control, but the only way to use cryptography involved the use of a base station. "Random Key Predistribution Schemes for Sensor Networks" by Chan et al.<sup>18</sup> had an analysis of using subgraphs of varying sizes that changed the probability of corruption by an adversary. Commercial vendors were interested in using small devices for customer authentication, but the processors could not carry out the public key computations that websites were beginning to use. In "A Practical Revocation Scheme for Broadcast Encryption Using Smart Cards" by Kogan et al.,<sup>19</sup> there was a way to revoke predistributed keys for these early access-control devices.



**Commercial vendors were interested in using small devices for customer authentication, but the processors could not carry out the public key computations that websites were beginning to use.**



**D**ave Wagner was the program committee chair for the 2003 Symposium. He made the following comments:

Reviewing the proceedings of the 2003 IEEE Security and Symposium, we see a number of themes, including interest in anonymous/private communication, network security, and a hope that network mechanisms could defend against DDoS, keeping track of the consequences of trust decisions, and detecting attacks.

What is striking is how prescient researchers were in anticipating problems that the field would struggle with. The impact of the solutions proposed varies. In retrospect, some of these problems got solved in different ways than researchers anticipated or weren't as serious as originally anticipated. We never solved DDoS in the network; instead, we scaled up the networks and scaled up servers (through replication) beyond the attackers' capacity. Instead of trying to detect software compromise with anomaly detection, we've focused on hardening software to make compromise harder. Sensor networks have ended up with fairly simple deployment models that haven't needed sophisticated key management schemes. Evasion attacks on intrusion detection, where intruders try sophisticated strategies to avoid being detected by the IDS, haven't turned out to be a serious issue. Most attackers don't bother, and a more critical focus is to detect attacks in the first place.

Some papers had a particularly significant impact or anticipated future trends. The work of Govindavajhala and Appel is an early precursor to modern-day work on hardware-level attacks, such as Rowhammer. The work on anonymous communication arguably influenced systems like Tor, which has had a significant impact. The line of work on puzzles and outsourced computation in the research literature can be loosely seen as one element that influenced the design of Bitcoin and Ethereum and more sophisticated blockchain systems.

At the time of the 2003 Symposium, *IEEE Security & Privacy* had just been launched. The goal of the magazine's organizing committee was to build a community of professionals spanning research and practice in information technology security and privacy. By 2003, it was clear that there was a worldwide need for accelerating security and privacy research and technology development. The magazine has been one facet of building the field's community of researchers and practitioners.

As we celebrate the 20th anniversary of the magazine, it is important to continuously enhance the relationship between the magazine and its namesake, the IEEE Security and Privacy Symposium, so that we cast a wide net across the larger community. ■

## References

1. X. Wang and M. K. Reiter, "Defending against denial-of-service attacks with puzzle auctions," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 78–92, doi: 10.1109/SECPRI.2003.1199329.
2. A. Yaar, A. Perrig, and D. Song, "Pi: A path identification mechanism to defend against DDoS attacks," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 93–107, doi: 10.1109/SECPRI.2003.1199330.
3. T. Yu and M. Winslett, "A unified scheme for resource protection in automated trust negotiation," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 110–122, doi: 10.1109/SECPRI.2003.1199331.
4. N. Li, W. H. Winsborough, and J. C. Mitchell, "Beyond proof-of-compliance: Safety and availability analysis in trust management," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 123–139, doi: 10.1109/SECPRI.2003.1199332.
5. D. Szajda, B. Lawson, and J. Owen, "Hardening functions for large scale distributed computations," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 216–224, doi: 10.1109/SECPRI.2003.1199338.
6. L. Zheng et al., "Using replication and partitioning to build secure distributed systems," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 236–250, doi: 10.1109/SECPRI.2003.1199340.
7. D. W. Price, A. Rudys, and D. S. Wallach, "Garbage collector memory accounting in language-based systems," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 263–274, doi: 10.1109/SECPRI.2003.1199342.
8. J. S. Shapiro, "Vulnerabilities in synchronous IPC designs," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 251–262, doi: 10.1109/SECPRI.2003.1199341.
9. S. Govindavajhala and A. W. Appel, "Using memory errors to attack a virtual machine," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 154–165, doi: 10.1109/SECPRI.2003.1199334.
10. D. Lie, J. Mitchell, C. A. Thekkath, and M. Horowitz, "Specifying and verifying hardware for tamper-resistant software," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 166–177, doi: 10.1109/SECPRI.2003.1199335.
11. D. Agrawal, D. Kesdogan, and S. Penz, "Probabilistic treatment of MIXes to hamper traffic analysis," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 16–27, doi: 10.1109/SECPRI.2003.1199324.
12. M. Wright, M. Adler, B. N. Levine, and C. Shields, "Defending anonymous communications against passive logging attacks," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 28–41, doi: 10.1109/SECPRI.2003.1199325.
13. G. Danezis, R. Dingledine, and N. Mathewson, "Mixminion: Design of a type III anonymous remailer protocol," in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 2–15, doi: 10.1109/SECPRI.2003.1199323.
14. U. Shankar and V. Paxson, "Active mapping: Resisting NIDS evasion without altering traffic," in *Proc. Symp.*

*Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 44–61, doi: 10.1109/SECPRI.2003.1199327.

15. H. H. Feng, O. M. Kolesnikov, P. Fogla, W. Lee, and W. Gong, “Anomaly detection using call stack information,” in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 62–75, doi: 10.1109/SECPRI.2003.1199328.
16. M. Backes and B. Pfitzmann, “Intransitive non-interference for cryptographic purposes,” in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 140–152, doi: 10.1109/SECPRI.2003.1199333.
17. D. Balfanz, G. Durfee, N. Shankar, D. Smetters, J. Staddon, and H.-C. Wong, “Secret handshakes from pairing-based key agreements,” in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 180–196, doi: 10.1109/SECPRI.2003.1199336.
18. H. Chan, A. Perrig, and D. Song, “Random key predistribution schemes for sensor networks,” in *Proc. Symp. Secur. Privacy*, Berkeley, CA, USA, 2003, pp. 197–213, doi: 10.1109/SECPRI.2003.1199337.
19. N. Kogan, Y. Shavitt, and A. Wool, “A practical revocation scheme for broadcast encryption using

smartcards,” *Proc. ACM Trans. Inf. Syst. Security*, vol. 9, no. 3, pp. 325–351, Aug. 2006, doi: 10.1145/1178618.1178622.

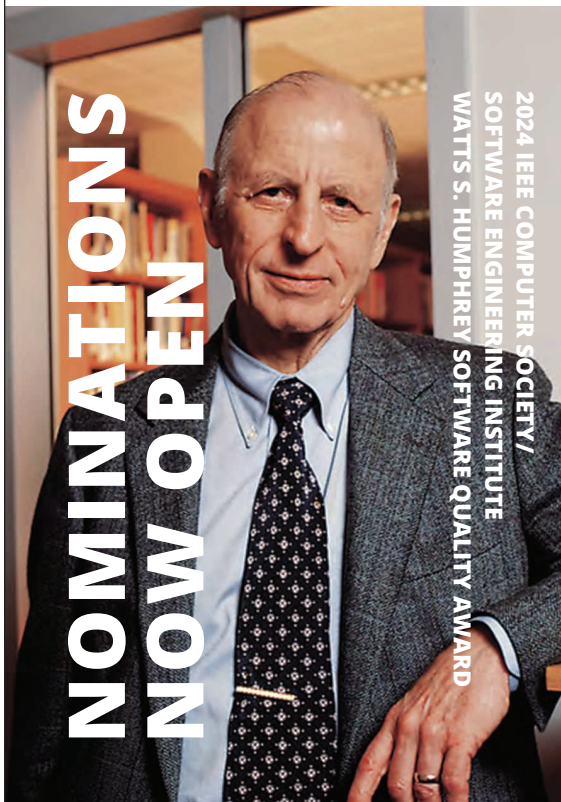
---

**Terry Benzel** is the director of networking and cybersecurity research at the Information Sciences Institute of the University of Southern California, Los Angeles, CA 90292 USA. Her research focuses on experimental cybersecurity. Benzel received an M.A. from Boston University and an executive M.B.A. from the University of California, Los Angeles. She is a senior member of the IEEE Computer Society, an associate editor in chief of *IEEE Security & Privacy*, and a member of the Board of Governors of the IEEE Computer Society. Contact her at [tbenzel@isi.edu](mailto:tbenzel@isi.edu).

---

**Hilarie Orman** is the president of Purple Streak. She consults and writes about security issues in computers, networks, and quantum computers. Contact her at [hilarie@purplestreak.com](mailto:hilarie@purplestreak.com).

## Carnegie Mellon University Software Engineering Institute



Since 1994, the SEI and the Institute of Electrical and Electronics Engineers (IEEE) Computer Society have cosponsored the Watts S. Humphrey Software Quality Award, which recognizes outstanding achievements in improving an organization’s ability to create and evolve high-quality software-dependent systems.

Humphrey Award nominees must have demonstrated an exceptional degree of **significant**, **measured**, **sustained**, and **shared** productivity improvement.

**TO NOMINATE YOURSELF OR A COLLEAGUE, GO TO**  
[computer.org/volunteering/awards/humphrey-software-quality](https://computer.org/volunteering/awards/humphrey-software-quality)

*Nominations due by September 1, 2023.*

### FOR MORE INFORMATION

[resources.sei.cmu.edu/news-events/events/watts](https://resources.sei.cmu.edu/news-events/events/watts)