Throughout modern society, considerable activity occurs on the Web. This activity is significant not only because of its volume, but also because of the variety of its participants. We all know someone whose parent or grandparent has an email address or a Web site, as well as someone whose five-year-old has used the Internet. In general, though, the type of Web activity is fairly limited. Most people go online to

- surf (use the Internet as an information resource),
- exchange email (use the Internet as a communications infrastructure),
- pass leisure time (use the Internet as an entertainment medium), or
- conduct financial transactions (use the Internet as a shopping or banking facility).

However, researchers are attempting, to take the Internet to the next level of importance in people’s lives. We can think of this as creating Web-based environments—communities of interest centered around a specific sphere of activity, where participants interact exclusively (or at least primarily) via the Web. These environments include all the buzz words that fill the media today, such as e-business, e-government, e-banking, e-law, e-learning, and e-health (see the sidebar for a concrete example).

Web surfing requires some minimal level of security, and online shopping requires a somewhat higher level of security. But participation in a Web-based environment can demand unprecedented levels of both security and privacy. This is why so much research activity currently exists in this area. Moreover, Web-based environments are complex and multifaceted; consequently, research into securing them is also complex and multifaceted. Thus, we might step back and put together a bigger picture of how this research is organized and interconnected.

As I mentioned, a Web-based environment is a Web-based community of interest. But a community doesn’t exist in a vacuum; it needs to live somewhere. If the goal is to build a community, then we need to create a structure—an edifice—in which this community can live and function. The construction of a physical structure (a house or office complex, for example) is a useful metaphor for the different subareas of research necessary for building secure Web-based environments.

**Construction phases**

There are six main phases in building construction: foundation, frame, exterior shell, plumbing and wiring, drywall, and finishing.

After these phases comes the final step of furnishing and decorating. Each phase parallels an area of Web-based-environment research.

**Foundation: Strong cryptographic algorithms**

Without a solid foundation, no building will last, and the same is true for the security tools and techniques needed to construct a Web-based environment. Over the past three decades several strong symmetric, asymmetric, hash, and MAC algorithms have been developed. We can always do more research (particularly as new cryptanalytic methods show some success), but the foundation in this area currently looks relatively strong.

**Frame: Strong cryptographic mechanisms**

In a physical building, the frame provides a base on which to work; it extends the foundation in a way that makes the rest of the construction possible. In security research, cryptographic mechanisms play this same role. Some common examples include cryptographic pseudorandom number generators, modes of operation for symmetric ciphers, challenge–response protocols for entity authentication, and key-establishment protocols that let two...
entities agree on what cryptographic key to use.

**Exterior shell: Secure communications protocols**

In a building, the exterior shell provides some (minimal) protection against the elements, but it isn’t enough to make the building habitable. Similarly, secure communications protocols protect data in transit but don’t offer enough protection to secure the whole environment. Protocols—largely developed by the Internet Engineering Task Force (IETF; www.ietf.org)—such as Internet Protocol Security (IPsec), Secure Sockets Layer / Transport Layer Security (SSL/TLS), Simple Public-Key GSS-API Mechanism (SPKM), Secure Multipart Internet Mail Extensions (S/MIME), and Time Stamp Protocol (TSP), can be counted in this category.

**Plumbing and wiring: Full security infrastructures**

The plumbing and wiring phase outfits a building for real use—most people would be unwilling to live in a house without running water or electricity. In the same way, security infrastructures make a Web-based environment usable. Such plumbing and wiring includes an authentication infrastructure (possibly a public-key infrastructure [PKI], although other authentication infrastructures are suitable for many environments), an authorization or access-control infrastructure, a privacy infrastructure, an infrastructure for malware protection (incorporating technologies such as firewalls, antivirus tools, intrusion-detection systems, and so on). With plumbing and wiring, we must remember that mistakes here can cause extensive, sometimes irreparable, damage to the building. Similarly, mistakes in implementing or installing these infrastructures can have serious security consequences; implementation or configuration blunders can easily expose an entire server or collection of machines to an adversary, for example.

**Drywall: Securing Web-based data**

The drywall phase in construction represents a major transition in the building’s progression (even if, in some ways, it’s only a perceptual difference). At this point, both the structure’s layout and the path forward suddenly seem very clear. For Web-based environments, this transition occurred when researchers began to realize that securing Web data using its own format and syntax could lead to some significant processing and representational advantages. The lingua franca of the Web is XML; thus, a few years ago, researchers began looking into ways to secure XML data using XML syntax. Technologies such as XML Encryption and XML Signature were the initial and fundamental moves in this direction, but many others soon followed, including Security Assertion Markup Language (SAML) Assertions, Extensible Access Control Markup Language (XACML), and WS-Security (www.oasis-open.org/committees/tp_home.php?wg_abbrev=wss).

**Finishing: Securing Web-based transactions**

The finishing phase makes a building ready for occupation and use. Note, however, that it’s “ready” in a generic sense; at this point, anyone could occupy and use it. For a Web-based environment, this phase corresponds to the research involved in securing the transactions that occur over the Web. The earliest and most well-known work in this area was defining HTTP over SSL (HTTP-S), but researchers have done much subsequent work, including SAML Protocols, XML Key Management Specification (XKMS), and security for Web Services Description Language/Universal Description Discovery and Integration (WSDL/UDDI).

**Furnishing and decorating**

The final step—furnishing and decorating—personalizes a building for a particular occupant. With respect to Web-based environments, this step corresponds to securing various aspects of the environment itself specifically for a given community’s needs. Examples of this area of research include the security services defined and specified in e-business XML (ebXML), Business Transactions Protocol (BTP), Web Services Business Process Execution Language (WSBPEL), Universal Business Language (UBL), LegalXML, eGov, and Election Services (see www.oasis-open.org for a more complete list).

**Contractors and subcontractors**

This analogy of the phases of build—
In several cases, the standards committee is the crucible from which emerges original research that lays the foundation for a new technology.

environments, and we can carry it even further. In construction, many contractors and subcontractors specialize in particular areas of expertise and don’t really see or actively participate in other parts of the work. Similarly, most researchers specialize in a relatively small number of areas (symmetric cipher design, PKI, or SAML, for example) and might not be fully aware of how and where their work contributes to the larger goal of building secure Web-based environments. The construction analogy might benefit such researchers by helping them see where their efforts are used in the overall project.

The construction site
Historically, research in many branches of science and engineering occurred in purely academic environments, such as universities and government-funded laboratories. Over time, a shift occurred, with considerable applied research happening in private industry offices—large and small companies were the birthplaces of major developments in many computer science and information technology fields, for example. More recently, however, another shift has occurred: in several cases, the standards committee is the crucible from which emerges original research that lays the foundation for a new technology. Once purely a way to achieve consensus on how to use an existing technology, the standards committee has in many instances taken an innovative role, creating the technology about which it will forge a consensus. This has been true for many aspects of the Internet and the Web, and it’s certainly true for technologies used for securing Web-based environments.

A recent example of this is the XACML Technical Committee in Oasis (www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml). This TC began in April 2001 with the goal of creating the first access-control-policy language fully specified in XML syntax. The participants (typically 10 to 20 committee members), who represent highly diverse segments of industry, government, and academia, recognized from the outset that countless proprietary and application-specific access-control-policy languages already existed (along with those specified in several standards bodies). However, they felt not only that an XML-based policy language for access control would benefit many environments, but also that they still needed to overcome significant technical challenges. Thus, discussions in the XACML TC quickly became innovative and led to some original research contributions. Examples of these contributions include:

- **Distributed policy authorship and reconciliation.** Many real-world environments have multiple policy-writing entities, which might be geographically distributed and which write policies autonomously and independently. However, when an access-control decision engine must make a decision regarding access to some protected resource, these policies might all simultaneously be relevant and will therefore need to be combined to facilitate correct decision-making. In some cases, these policies might conflict, so the decision engine will need to know how to resolve this conflict. For example, should corporate policy always override departmental policy, or should some other rule apply? XACML specified the syntax to refer to a policy-combining algorithm so that a higher-level policy (a policy set) can explicitly say how to unambiguously combine and reconcile a collection of related policies.

  - **Policy applicability versus policy evaluation.** In many large environments, hundreds or thousands of valid policies may exist. However, only a very small subset of these might be relevant to an access-control decision for a specific resource. For efficiency reasons, it’s important to separate the process of determining whether a candidate policy is applicable to a given access-control decision from the process of actually evaluating the policy to render the decision. In previous efforts, these weren’t cleanly separated; applicability was determined only by evaluating the full policy. In XACML, a policy is conceptually split into two pieces—one allows a very computationally efficient determination of applicability, and the other is the more detailed and arbitrarily complex Boolean logic facilitating full-policy evaluation.

  - **Content introspection.** When rendering a decision, the access-control decision engine typically compares the subject’s attributes with the resource’s attributes according to some rule (the classical military example is that the subject’s clearance must equal or exceed the data’s classification for read-access to be granted). With XML policies, however, the resource to be protected is itself XML data. Thus, we...
can often find resource attributes within the resource, rather than stored separately using some other mechanism (such as an attribute certificate). XACML TC members explored and explicitly incorporated the notion of content introspection, whereby policy rules can be based on internal resource data that might be found at runtime by using XPath technology.6

These and other research contributions resulted directly from the collaboration of committee members in the context of the XACML TC; they didn’t come from elsewhere simply for discussion and incorporation into the specification. Hammering out the XACML specification was, at times, a collaborative research effort (that is, not just a consensus-reaching exercise), identical in many ways to the research collaborations you used to find only in universities and government labs.

When building secure Web-based environments, the construction site is increasingly found in teleconferences and hotel meeting rooms. Furthermore, the construction workers are an international group of highly motivated individuals, often with interoperability on their minds.

**When can we move in?**

We have the construction site and the construction workers, we know the necessary construction phases, we have a good idea of what to build, and we have many of the necessary tools. The obvious question is: When can we move in? When will these secure Web-based environments be ready for use? We’re getting close, but it might still take awhile.

In all the construction phases I mentioned, significant research and development has already occurred. However, in each phase, considerable opportunity exists for further work. At the algorithmic level (phase 1), cryptanalytic techniques continue to improve; algorithms thought to be strong are broken or are shown to be weaker than expected with every new conference or journal issue. At the mechanism level (phase 2), research work continues on modes of operation for symmetric ciphers, as well as fundamental areas such as good pseudo-random number generation. With secure communications protocols (phase 3), many choices are available, but few have seen widespread use and none are ubiquitous (although Secure Sockets Layer is close to ubiquitous; its replacement, TLS, hasn’t been implemented quite as widely). At the infrastructural level (phase 4), we must overcome many problems, particularly those of efficiency, practicality, implementation, configuration, and getting the underlying trust model right. The drive to bring security to Web data and transactions by incorporating it into XML data structures (phase 5) and protocols (phase 6) is relatively recent; much has to be solved in these two areas. Finally, the “furnishing and decorating” step of personalizing security for particular Web-based environments is also fairly recent; many problems this area must address will only become evident as these environments are more fully designed, developed, deployed, and used.

Building secure Web-based environments is a stimulating, exciting, and complex research area. Furthermore, it’s interdisciplinary (computer scientists, engineers, mathematicians, language experts, distributed systems experts, and many others are all involved in various aspects of the problem) and collaborative (the “construction workers” include academics, large and small businesses, competitors, and noncompetitors, as well as people in law, government, healthcare, and education, among other fields). Most of all, however, it’s real: A strong desire exists to use the Web to do things that have never been done before (for cost reduction, efficiency, and usability reasons), and a genuine sense of urgency exists for getting things working. We have a lot to do, but many significant accomplishments have already occurred, and the future looks very promising. □

**References**


Carlisle Adams is an associate professor in the School of Information Technology and Engineering (SITE) at University of Ottawa, where he is director of the Information Security Research Group. His research interests include cryptography, computer and network security, access control, and privacy. He has an MSc in computer science and a PhD in electrical engineering from Queen’s University and is licensed as a professional engineer (PEng). He is a member of the ACM and the International Association for Cryptologic Research (IACR), and a senior member of the IEEE. Contact him at cadams@site.uottawa.ca.