Missing Link
Security Services
Mark Bouchard, Founder

A Security Strategy for Web 2.0 and Social Networking

About the Author

Mark Bouchard, CISSP, is the founder of Missing Link Security Services LLC, a consulting firm specializing in information security and risk management strategies. A former META Group analyst, Mark has assessed and projected the business and technology trends pertaining to a wide range of information security topics for over 10 years. He is passionate about helping enterprises address their information security challenges. During his career he has assisted hundreds of organizations worldwide with strategic and tactical initiatives alike, from the development of multi-year strategies and overall architectures to the justification, selection, acquisition, implementation and operation of individual security and privacy solutions.

Introduction

The onslaught of Web 2.0 and social networking is essentially unstoppable. Many enterprises are embracing related technologies and solutions to take advantage of promised benefits, such as efficient collaboration and greater adaptability to rapidly changing business plans. But even for shops trying to avoid it, resistance is essentially futile. Web 2.0 and social networking applications are already ingrained in our culture and, as a result, employees expect to use them whether or not such activity is formally sanctioned. Regardless of the actual corporate strategy, therefore, IT must be prepared to deal with the consequences – especially when it comes to maintaining adequate levels of information security and privacy.

This paper will help organizations understand the nature and impact of the phenomenon that is Web 2.0 and social networking. Applicable security challenges will be explored and a concise yet comprehensive strategy will be provided for mitigating against them.

Web 2.0 … in a Nutshell?

To be perfectly clear, it's not actually possible to fit Web 2.0 into a nutshell. Unfortunately, it lacks a simple or widely accepted definition and has so many facets that it routinely means radically different things to different people. That said, a definition of sorts can be fashioned by looking at Web 2.0 along three inter-related dimensions: its fundamental principles, representative sites/applications, and underlying technologies.

Chief among the principles that characterize Web 2.0 is the notion of "the Web as a platform." At a high level, this captures the goal of building applications to take advantage of the inherent features and strengths of the Web, versus simply using the Internet as a means for transport. Far more relevant from the perspective of security and privacy, however, are some of the derivative conditions that apply. For instance, with Web 2.0:

- The packaged software model and thick-client implementations give way to web applications that are deployed and consumed as hosted services;

- Unilateral content generation gives way to an architecture of participation;

- Data, instead of being constrained to pre-determined uses/applications, becomes a "remixable" resource; and,

- Static, manual driven interactions and transactions give way to a more dynamic, "rich user experience."

Taking a look at the second dimension, the categories of applications that embody these principles include blogs, wikis, social networking, RSS feeds, mashups, software as a service, and rich internet applications. More specifically, representative sites (= applications) include Wikipedia, Facebook, LinkedIn, and Google Apps. Finally, there's the alphabet soup of underlying technology, including AJAX, Flash, XAML, REST, XML, JSON – not to mention "conventional" Java and Active-X browser plug-ins.

The key points to take away at this stage are: (a) that Web 2.0 is not one "discrete" item, rather it's a large and still expanding collection of techniques, technologies, and applications; and (b) that Web 2.0 involves a fundamentally different and more open way of generating, sharing, and processing data.

No Way Out

Another crucial aspect that needs to be acknowledged is that Web 2.0 is inescapable. Periodic polls routinely reveal a non-trivial percentage of managers that consider it unnecessary to implement relevant countermeasures because their organizations have yet to employ any Web 2.0 solutions. However, this is a risky way to think – not to mention operate. The fact of the matter is that Web 2.0 applications are already present on the majority of corporate networks, whether they've been formally/centrally approved or not. Furthermore, the number of organizations where Web 2.0 has yet to be embraced as a legitimate business tool is dwindling fast. Just consider these findings:

- 42% of office workers between the ages of 18 and 29 discuss work-related issues on blogs and social networking sites (YouGov);

- 50% of surveyed organizations indicate that at least 30% of their network bandwidth is being consumed by social networking traffic (Forrester);

- it is estimated that nearly half of all web developers are already using AJAX;

- more than 30% of large companies will have a Web 2.0 business initiative underway by 2008 (Gartner); and,

- 66% of surveyed organizations indicate that Web 2.0 is essential to maintaining their company's market position (McKinsey).

A Closer Look at the Problem

Given the current penetration and expected growth rates for Web 2.0 solutions, it is certainly appropriate to establish a better understanding of the related security challenges as a prerequisite to doing something about them. This has already been alluded to, but for many organizations one aspect of the problem will be the need to overcome the misconception by some executives that the lack of a Web 2.0 strategy equates to Web 2.0 applications not being present in the corporate computing environment. The balance of the significant issues can be categorized by the various features that characterize Web 2.0 and social networking solutions, as indicated below.

- The *pervasive accessibility* of Web 2.0 applications – which are typically externally hosted and can be accessed from virtually any location and/or device – erodes the effectiveness of common, DMZ-centric security strategies.

- The *participatory nature* of these solutions raises the exposure and susceptibility of users to social engineering techniques and further de-sensitizes them when it comes to clicking on links. Even more worrisome is the potential for hackers to "contribute" content that incorporates malware and/or links to infected sites, as well as the growing ease with which sensitive information can be inadvertently or intentionally "leaked."

- Rich internet applications are a veritable hornet's nest of risks. For starters, there is no way around the fact that *new technology is essentially vulnerable technology:* early in a technology's lifecycle developers inevitably emphasize business functionality over security; the hosted model enables new code/capabilities to be deployed as soon as they are developed, creating a "perpetual beta"; organizations simply have insufficient operational experience with Web 2.0 technologies and, therefore, a limited understanding of how to protect them; and, the market in general has yet to provide much in the way of security capabilities/products that are focused on providing Web 2.0 protection.

- It also doesn't help that the *underlying technology is relatively complex.* AJAX is not just one thing (i.e., JavaScript); rather it is a framework for communication that can accommodate a wide variety of protocols and data formats – all of which need to be understood in order to be secured.

- The *asynchronous communication techniques* often associated with having a rich user experience can facilitate automatic/dynamic execution of code (including malware) on client devices without user involvement or knowledge.

- *Input validation is complicated* both by the "architecture of participation" and the reusable, remixable nature of data sources (especially in the case of mashups).

Of course, the above list only hits the highlights. There are many other issues that will ultimately require consideration as well such as: the potential for secure development lifecycle processes to break down as users/groups take advantage of ubiquitous toolkits and cobble together new Web 2.0 applications outside the influence and oversight of IT; or the question of legal rights and liability when re-using content from external sources, either from the perspective of protected property or the promulgation of "pass-thru threats." Regardless of the details, however, it is absolutely critical for organizations to accept the over-riding reality that Web 2.0 is elevating web traffic (i.e., HTTP and HTTPS) to exceed email (i.e., SMTP) in terms of being a primary vector for threats against their computing environment.

Divide and Conquer

The scope of the security challenges pertaining to Web 2.0 and social networking is certainly formidable. To avoid being overwhelmed, it is recommended that organizations break the overall problem of establishing adequate protection into a handful of smaller, more manageable tasks. Along these lines, one very practical approach is to develop separate solutions to address the needs of different constituencies – in other words: to protect local users as consumers of Web 2.0 applications, to protect remote/mobile users as consumers of Web 2.0 applications, and to protect the enterprise itself as a host/provider of Web 2.0 services.

Security for Local Users

A concise strategy for protecting local users (i.e., employees and guests) as consumers of externally available Web 2.0 services entails three core elements. Enterprises must first address the basics. Appropriate use policies should be extended to explicitly cover Web 2.0 and social networking, as should corresponding training materials that foster user awareness.

Next, enterprises need to address the potential for data leakage. Ideally, attempts should be made to get the proverbial horse back in the barn by finding and removing sensitive information from locations where it shouldn't be. Subsequently, the goal is to keep it in the barn, for example by (a) more thoroughly controlling access to data in the first place, and (b) monitoring for it at well known exit points using functionality associated with an existing email security gateway, or even a dedicated data leak prevention solution.

Finally, an analog to the email security gateway should be deployed to provide a wide range of countermeasures that better address the rising potential of web traffic as a threat vector, including: enforcement of user and application-specific access controls, URL filtering, and anti-malware filtering for HTTP (and ideally HTTPS) that is based not just on signatures but that incorporates real-time analysis/detection techniques as well. Either a dedicated web security gateway or a UTM device will typically do the trick – although the latter can have several advantages. These include the ability to address other requirements if needed (e.g., network-level firewall/protection, email security, and VPN), as well as the potential to simplify an organization's overall network and security architecture by reducing the number of different products needed to provide comprehensive technical coverage.

Security for Remote/Mobile Users

At a high level, the strategy for protecting remote/mobile users is basically the same. Ideally, all of the same policies, practices and mechanisms should be put in place. The challenge in this case, however, is that the users are not conveniently located within the physical boundaries of the enterprise or, quite possibly, not even in a single, fixed location. Still, there are a handful of viable and emerging options that can be pursued, each with its own set of tradeoffs.

The thick-client approach entails loading corporate-managed laptops with client-based equivalents of most, if not all, of the technical countermeasures outlined in the previous section – plus some software to facilitate secure, remote configuration management. This approach generally yields the greatest degree of visibility and control, but is also the most expensive and difficult to manage.

The "tunnel all" approach is definitely less costly, but runs the incremental risk that it can be circumvented by crafty users. In this case, a secure remote access solution, such as SSL VPN technology, is used to route ALL sessions back to a corporate site. Corresponding traffic can then take advantage of gateway based countermeasures, just as if the users were operating locally.

The thin-client approach may also take advantage of secure remote access technology. In this case, however, the user's machine is little more than a "dumb terminal" used to connect to a virtualized desktop hosted in the organization's data center. This way all of the user's work-related sessions are protected by countermeasures that are reliably deployed and managed back at headquarters. And even if they connect directly to the Internet and conduct unsanctioned activities, the risk is minor since their machine retains no corporate data and has a minimal software footprint. Of course, working "offline" is not possible in this scenario.

It's also worth noting that an SSL VPN solution could be used in all three cases to help facilitate secure deployment of associated software elements and to enforce/manage which users are required to use which approach (e.g., via associated host integrity checking capabilities).

Security for the Enterprise

The final area to be addressed is the need to protect the enterprise as a provider/host of Web 2.0 and social networking services. In general, protecting intranet implementations should be fairly straightforward. After all, the organization retains complete control over both ends of the interaction (i.e., client and server). That leaves extranet scenarios. The countermeasures that should be considered for this type of implementation include the following.

- Particular care should be taken when selecting packaged Web 2.0 applications and Web 2.0 development toolkits. Emphasis should be placed on assurances and certifications pertaining to security and quality control and, more importantly, on having a proven track record in these areas.

- Secure coding practices are essential, as is Web 2.0-specific security training for developers. Special attention should be paid to the practice of validating inputs.

- Hacker tools should be deployed internally, especially fuzzers and Web 2.0-capable application scanners.

- In addition to deploying a conventional Web/SOA/XML firewall, it may also make sense to run a UTM device or web security gateway in front of hosted Web 2.0 services. Operating such a solution essentially in reverse could help block content "contributions" that include malware and/or malicious links, thereby preventing harm to other users and damage to the organization's reputation that could limit the usefulness and value of its offerings.

Conclusion

As with some of the user/consumer-focused technologies that preceded Web 2.0 and social networking solutions – such as email, instant messaging, and video – organizations that choose to ignore them do so at considerable risk. Because they have rapidly become a staple of our Internet culture, Web 2.0 applications will be present on corporate networks whether they are formally embraced or not. Moreover, there is virtually unlimited potential for creative and constructive Web 2.0 implementations that yield significant business value. Either way, it is imperative that today's IT organizations gain a better understanding of the associated security and privacy risks, and proactively develop a comprehensive strategy to address them. Some of the specific technical countermeasures that will undoubtedly prove useful in this regard include web security gateways, SSL VPNs, and UTM devices – especially if they have Web 2.0-specific visibility and control capabilities.