

Small Business Checklist for Evaluating an ASP

	Step	Reason	Tools/References
1.	Review information on application. All printed materials, sales documents, and contact information.	Provides background for evaluating function of app for business.	
2.	Summarize what the application will be used for, how it will be used, and by whom. Specify what information the ASP holds.	Provides summary for evaluation/report to management. Basis for rating the criticality of the data at the ASP.	
3.	Contact the application developer or company representative to establish testing boundaries and get written permission for testing from them before any actual testing is done.	Written permission for testing is absolutely necessary. Be prepared to outline which tools you will be using—and what their effect on the application may be.	
3a.	If possible, get a separate admin and user account strictly to use for testing.		
3b.	Ask for any policies related to server patching—who watches for new vulnerabilities in this company.	Open ended questions can provide a good foundation for an evaluation	
3c.	Ask for policies related to passwords	Length, pw history, complexity—how does their application handle passwords?	
3d.	Ask for any third party security certification documentation and reports.	Have they been evaluated/certified by an outside company?	

3e.	Ask for general information about firewall/perimeter protection.	How is their application structured? Web server in a DMZ/ db server on a trusted net? Or everything on one box? Are other websites hosted on the same server?	
3f.	Ask for general information about application level protection.	Do they use an application level firewall?	http://www.owasp.org/development/codeseeker
3g.	Ask about logging/auditing of application-do they log IP info, username/pw info, time of day information. How long are logs kept?	Will you be able to glean information from these logs in the case of a security incident?	
3h.	Explain in detail the process <your company> uses to evaluate an application/server.	Inform them well so that there are no surprises when you are testing... it may be a good idea to list your tools if they seem reluctant or hesitant.	
3i.	Determine the level of QA and code review.	Do they have a process/personnel for QA testing and code review- is it the people who are responsible for developing the code?	
3j.	Ask about insurance coverage.	Do they carry "cyber insurance" that would provide coverage for security related events?	
3k.	Ask what their process is in regards to California Law SB 1386.	Do they encrypt all data in all databases? If you have customer identifiable information and have California customers, what would be the ASP's process for notification? Are they prepared to assist with notifications?	http://info.sen.ca.gov/pub/01-02/bill/sen/sb_1351-1400/sb_1386_bill_20020926_chaptered.html
3l.	Confirm the web-server OS and server software/version.	Verify this in the testing phase.	

3m	Ask for any additional information that the developer or company representative may provide that would be helpful in evaluating the application	Open ended questions – let them talk about their application and network environment.	
4	Review provided documentation to establish auditable items.	Create an application specific checklist from the information provided to you.	
5.	Web Server FQDN and IP Address	Identify the specific server you'll be testing.	-NSLookup (online tool) http://network-tools.com/nslook/ -WHOIS information http://www.networksolutions.com/cgi-bin/whois/whois
6.	Network testing—port scan	See what is open to the Internet—is it just ports 80 and 443? (HTTP and HTTPS) What else is open?	-GFI LANscanner http://www.gfi.com/lannetscan/ -nMap http://www.insecure.org/nmap -nMapWIN http://mypage.bluewin.ch/vogje01/e/nmapwin/index.html
7.	Site Map	Will enable you to view/search source code for sensitive information: <ul style="list-style-type: none"> • hidden • <!— • NAME=GENERATOR • METHOD=GET • Copyright • \ • / <p>Are there any third-party products used? Have known defaults for these products been tested?</p>	-Achilles http://www.mavensecurity.com/achilles -Black Widow http://www.softbytelabs.com
8.	Webserver and OS versions	Revealed in headers—can view in Achilles logfiles Is this information aligned with what you discovered in the interview process in step 3 ?	Online tool http://www.netcraft.com

9.	Authentication and encryption	<p>Is SSL configured correctly-is it user friendly?</p> <p>List certificate related browser warning, if any.</p> <p>Any pages containing a mix of encrypted/plaintext data?</p> <p>Document all SSL ciphers allowed by site.</p>	<p>Use "What's that SSL site running?" on Netcraft</p> <p>CTR-I using Netscape browser will provide encryption info</p>
10.	Sign-on Issues	<p>Friendly error messages?</p> <p>Can accounts be brute-force attacked?</p> <p>Can passwords be harvested?</p>	<p>Webcracker 4.0</p> <p>http://packetstormsecurity.nl/Crackers/indexsize.shtml</p>
11.	Session-level Issues	<p>Does the site allow concurrency?</p> <p>How long is the inactivity timeout?</p>	
12.	Other security issues- this step MAY be optional because of the nature of the tool.	<p>Nikto performs a comprehensive, fairly obvious scan-if you want to use Nikto on the ASP site, MAKE SURE you describe your process and the tool in detail to the people responsible for the site.</p>	<p>NIKTO http://www.cirt.net/code/nikto.shtml</p>
13.	Transaction-level- from mirrored site info	<p>Where are hidden form elements used? Does manipulating them adversely affect the server?</p> <p>Document any server-generated error visible to a remote user.</p> <p>Where are GETS used for user input?</p>	<p>Odysseus</p> <p>http://www.wastelands.gen.nz/index.php?page=odysseus</p>