Technical Risk Analysis

Karl Cronburg
karl@cs.tufts.edu
Nov-11-2013
Comp 116 - Computer Security

Risk ID	Technical Risk	Technical Risk Indicators	Impact Rating	Impact	Mitigation	Validation Steps
1	PHP injection	apache access logs containing PHP code	Very High	Unrestricted access to arbitrary PHP execution - privacy & system performance	Validate user input via GET HTTP parameter	Cast $_GET[“id”] to an integer
2	SQL injection	Missing or altered SQL tables, odd log entries	High	Unrestricted access to SQL databases - privacy	Sanitize user input	use mysqli_real_escape_string
3	Hard-coded Passwords	Unaccounted for logins to an account	Medium	Access to user accounts or SQL databases - privacy & system performance	Store passwords outside of code, in a secure locn	Set proper options in config files
4	XSS - can inject HTML tags into a web page	Unexpected content in a web page	Medium	Arbitrary browser-side code execution - privacy & system performance	Validate user input	Compare data with form of expected input
5	Information Leakage	die(“”) located in any PHP file	Low	Attacker gains information about server environment - privacy	Print less detailed error messages to end user	Remove usage of PHP die() when given system-specific information
6	Buffer Overflow	Network accessible programs regularly crashing, possible	Medium	Arbitrary execution of shell code on	Fix the buffer overflow, or use “least	Either fix the overflow, or sandbox the process in a

		DoS		your machine with same privilege as vulnerable program - system performance	privilege” to mitigate the problem	non-root / limited user account
7	Information Leakage	Increased web traffic, possible DoS	Low	Attacker gains information about server environment - privacy	Banner scrubbing	Delete version information from all network-enabled applications
8	Weak Authentication, passwords easily cracked	Increased web traffic, possible DoS	Medium	Access to user accounts - privacy & system performance	Password requirements	Require partially non-alphanumeric passwords
9	Single user system with only one account (root)	System performance degraded	Very High	Access to root account - privacy & system performance	Create a separate user account for each critical application	Linux-based users, groups, and permissions
10	Phishing	Increased email traffic	Medium	Most often privacy loss (tricking a user into entering	Spam filtering & email authentication	Emails from your domain (e.g. tufts.edu) should always be authenticated as being sent from authorized servers