



PENETRATION TEST

ACME CORP. BAT PORTAL PENTEST

WEBAPP1001 Saturday, May 24, 2025



	TESTING SUMMARY					
Start	Jan 14 2024	Tested	100.00%			
End	Jan 24 2024	In Progress	0%			
Completed	100%	Not Tested	0%			
Total Testcases	127	Not Applicable	0%			
UNIQUE VULNERABILITIES						
Total	3	3				
Critical	0	2				
High						
Medium	2	1				
Low	0					
Info	0	0 Critical	High Medium Low Info			







EXECUTIVE <u>SUMMARY</u>

Objective

The objective of testing was to assess the security posture for the new AttackForge Bat Portal, from the perspective of an external self-registered user.

Approach

The following scenario's were assessed:

- Attacker has self-registered an account on the AttackForge Bat Portal.
- Attacker has access to compromised staff user credentials for an account on the AttackForge Bat Portal.

Summary

AttackForge was able to compromise the application, and subsequently gain access to the internal corporate network. From here, AttackForge was able to gain access to credit card data within the Secure Payments Area (SPA) within Cardholder Data Environment (CDE).



🛞 Bluff (ity Golf853 - NCR Counterpoint											-8X
	4 counterpoint							C	2 🞯	* 🗵	MGR SYSADMI	IN •
		G	Home > Setup	> System > Col	nfiguration							
*	System	1	System Data Dictionary					- x	6	- 8 X		
₩a	Configuration	G	I ables IM v IM_item	Table Column Index Description liter	n				ŝa lo code	Goback		
6	Reports		IM_ITEM IM_ITEM_EC_DEL	Fast open mode Ye	4 ~		Gio to virtual	table	Key Required	v v		
iðj	Inventory	Ý	IM_ITEM_NOTE IM_ITEM_PROF_COD • Customize table	Use without table			Go back	k. ez	Keyword	~		
٦£	Customers	~	Columns	Add-on-the-fly form	~			~				
ជា	Purchasing	×	ADDL_DESCR_2 ADDL_DESCR_3	inter								
R	Point of Sale	~	ALT_1_DENOM ALT_1_NUMER	Custom maint form Custom view form			Edi					
0	Ecommerce	~	ALT_1_PRC_1 ALT_1_REG_PRC ALT_1_UNIT	Custom add-on-the-fly form Standard form	SimplifieditemAddOnThe	Flyaml	Ed/ Ed/	** Standard **				
+			ALT_1_WEIGHT ALT_2_CUBE ALT_2_DENOM	This table contains items - The table is virtual because	goods and services you have several views th	ve for sale. at look "almost like" items ((and users could possib	ly create		<u> </u>		
			ALT_2_NUMER Customize column	custom views that similarly	resemble items).							
48			IN_ITEM_X1 A									
			IM_ITEM_X3 IM_ITEM_X4 IM_ITEM_X5							¥ (>)		
			Eustomize index					9		Сорупс	ht © 2017 NCR	
	O Type here to search		4 🖻	🗮 🛱 🎯	1	1	E	^ 9	● 40) 転	<i>d</i> 🖿	ENG 9:42 AM 7/5/2017	\Box

Figure 1: Access to Credit Card Data in SPA



SUMMARY OF <u>RECOMMENDATIONS</u>

The following recommendations are made to <CUSTOMER> as a result of this assessment:

Recommendation 1: Do this thing..

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam ipsum augue, finibus id pretium sit amet, semper nec lorem. Nam bibendum arcu sed odio iaculis, a pretium eros ornare. Suspendisse in imperdiet ipsum, sed congue nisi. In ullamcorper feugiat bibendum. Vestibulum eu diam sed diam ultrices cursus eu vel nisl. Phasellus vestibulum est eu faucibus tempor. Integer id elementum enim, quis tristique est.

Recommendation 2: Do that thing..

Integer neque urna, elementum at nibh ut, ultrices pulvinar lacus. Donec eget turpis porttitor lectus laoreet euismod. Vivamus suscipit gravida metus vitae pellentesque. Ut maximus dictum mi, ut accumsan nulla maximus et. Vestibulum auctor quis nulla pulvinar eleifend. Aliquam aliquam iaculis est blandit dapibus. Sed posuere ipsum sed consectetur ultrices. Vestibulum feugiat vulputate magna eget commodo.



POSITIVE SECURITY OBSERVATIONS

The following positive security observations were observed during this assessment:

- Lorem ipsum dolor sit amet, consectetur adipiscing elit.
- Nullam ipsum augue, finibus id pretium sit amet, semper nec lorem.
- Nam bibendum arcu sed odio iaculis, a pretium eros ornare.
- Suspendisse in imperdiet ipsum, sed congue nisi.



SUMMARY FINDINGS

PRIORITY	VULNERABILITY	REMEDIATION STATUS
HIGH	Unrestricted Upload of File with Dangerous Type	\bigotimes
MEDIUM	Inconsistent Access Control	\otimes
MEDIUM	Relative Path Traversal	$\overline{\mathbf{x}}$





ATTACK<u>CHAINS</u>

1. <u>Gain control of core web</u> <u>server to further pivot attack</u> <u>into ACME Corp. internal</u> <u>network.</u>



External Attacker

Attacker who has self-registered account on ACME Corp. Bat Portal Internet-facing application.



Action

Log into application and enumerate vulnerable file-upload



Exploit High Vulnerability

Attacker identifies vulnerable upload functionality in MyProfile and uploads web shell.

Slide 9 of 29





Action

Search for ways to trigger web shell.





Exploit Medium Vulnerability

Attacker enumerates server directory structure to navigate directly to uploaded web shell.



Target Server

Attacker triggers web shell, elevates to full shell, then creates back door in web server for persistent remote access.



Captured Flag

Operating-System access to compromised ACME Corp. Bat Portal web server allowing further attack into ACME Corp. internal network.



VULNERABILITIES

1. Unrestricted Upl	oad of File with Dangerous Type	
	CVSSv3 SCORE	
Base	8.5	
Temporal	8.5	
Environmental	8.6	
Vector	CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H	
	DESCRIPTION	

<u>An unrestricted upload of files with dangerous type</u> often occurs in applications that explicitly trust application users will submit files of specific type and content only. Uploaded files are then either passed to other internal components for further processing or remain stored for future use within the application in an easily guessable location.



In a typical attack scenario, an attacker discovers the upload functionality in the application and submits a specifically crafted file that embeds malicious content, e.g. virus, exploit or shell code. From this point the attacker either passively waits until the malicious content is accessed and executed by an internal component, other system or user, or if the file is stored in a discoverable location the attacker tries triggering file execution within application by leveraging application mapping of known file types to specific execution routines. An insider or an attacker who can get administrative access to application can upload a web shell, then upload a normal shell and escalate privileges. The attack can be further propagated to other hosts on same network segment.

ATTACK SCENARIO

Arbitrary code execution is possible if an uploaded file is interpreted and executed as code by the recipient. This is especially true for .asp and .php extensions uploaded to web servers because these file types are often treated as automatically executable, even when file system permissions do not specify execution.





Figure 2

REMEDIATION RECOMMENDATION

- Assume all input is malicious. Use an 'accept known good' input validation strategy, i.e. use a whitelist of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does.
- 2. Generate a new, unique filename for an uploaded file instead of using the user-supplied filename, so that no external input is used at all.



- 3. Define a very limited set of allowable extensions and only generate filenames that end in these extensions.
- 4. Consider the possibility of XSS (CWE-79) before allowing .html or .htm file types.
- 5. Ensure that only one extension is used in the filename. Some web servers, including some versions of Apache, may process files based on inner extensions so that 'filename.php.gif' is fed to the PHP interpreter.
- 6. For any security checks that are performed on the client side, ensure that these checks are duplicated on the server side, in order to avoid CWE-602. Attackers can bypass the client-side checks by modifying values after the checks have been performed, or by changing the client to remove the client-side checks entirely. Then, these modified values would be submitted to the server.
- 7. Do not rely exclusively on the MIME content type or filename attribute when determining how to render a file. Validating the MIME content type and ensuring that it matches the extension is only a partial solution.
- 8. Do not rely exclusively on looking for malicious or malformed inputs (i.e., do not rely on a blacklist). A blacklist is likely to miss at least one undesirable input, especially if the code's environment changes. This can give attackers enough room to bypass the intended validation. However, blacklists can be useful for detecting potential attacks or determining which inputs are so malformed that they should be rejected outright.
- 9. For example, limiting filenames to alphanumeric characters can help to restrict the introduction of unintended file extensions.



	AFFECTED ASSET				
Open	<u>bat-portal.attackforge.com</u>				
	 Do this Do that 				
POC	<some script=""><do something=""></do></some>				



Uname: User: Php: Hdd: Cwd:	Linux lamp 4.15.0-4 33 (www-data) Gri 7.2.15-0ubuntu0.18. 15.68 GB Free: 9.2 /var/www/html/ drw	7-generic #50-Ubuntu : oup: 33 (www-data) 04.2 Safe mode: OFF 7 GB (59.09%) /xr-xr-x [home]	SMP Wed Mar 13 10:44:52 UTC 2 [phpinfo] Datetime: 2019-04-14	019 x86_64 [Google] [Exploit-DB] 22:02:49		UTF-8 () Server IP: 10.0.2.15 Cilent IP 192.168.56.1
[Sec.]	nfo] [Files] [Console] [Infect]	[Sql] [Php] [Safe mode] [String tools] [Bruteforce]	[Network] [Logout] [Self remove]
File r	manager					
Nam	ne –	Size	Modify	Owner/Group	Permissions	Actions
1		dir	2019-03-19 09:46:55	root/root	drwxr-xr-x	RT
	re]	dir	2018-03-07 21:10:20	www-data/www-data	drwxr-xr-x	RT
[dru	upal-8.5.0]	dir	2019-04-12 11:43:53	www-data/www-data	drwxr-xr-x	RT
- [mo	odules]	dir	2018-03-07 21:10:20	www-data/www-data	drwxr-xr-x	
[new	w_folder]	dir	2019-04-02 12:31:42	www-data/www-data	drwxr-xr-x	RT
C [pro	ofiles]	dir	2018-03-07 21:10:20	www-data/www-data	drwxr-xr-x	
C [site	es]	dir	2018-03-07 21:10:20	www-data/www-data	drwxr-xr-x	RT
🗌 [the	emes]	dir	2018-03-07 21:10:20	www-data/www-data	drwxr-xr-x	
C [ver	ndor]	dir	2018-03-07 21:23:44	www-data/www-data	drwxr-xr-x	RT
🗌 com	poser.json	2.68 KB	2018-03-07 21:10:20	www-data/www-data	-rw-rr	RTFED
com	poser.lock	157.30 KB	2018-03-07 21:10:20	www-data/www-data	-rw-rr	RTFED
📃 diy.p	hp	31 B	2019-04-04 21:43:03	www-data/www-data	-rw-rr	RTFED
hello	o.sh	18 B	2019-04-04 14:53:47	www-data/www-data	-rwxr-xr-x	RTFED
inde:	x.php	549 B	2018-03-07 21:10:20	www-data/www-data	-rw-rr	RTFED
LICE	ENSE.txt	17.67 KB	2016-11-16 23:57:05	www-data/www-data	-rw-rr	RTFED
REA	DME.txt	5.75 KB	2018-03-07 21:10:20	www-data/www-data	-rw-rr	RTFED
🗌 robo	ts.txt	1.56 KB	2018-03-07 21:10:20	www-data/www-data	-rw-rr	RTFED
🗌 simp	ble1.php	341 B	2019-03-22 10:21:01	www-data/www-data	-rw-rr	RTFED
📃 simp	ole2.php	112 B	2019-03-22 10:21:22	www-data/www-data	-rw-rr	RTFED
📃 simp	ole3.php	177 B	2019-03-22 10:21:37	www-data/www-data	-rw-rr	RTFED
web.	.config	4.45 KB	2018-03-07 21:10:20	www-data/www-data	-rw-rr	RTFED
wee	vely.php	669 B	2019-03-28 14:48:24	www-data/www-data	-rw-rr	RTFED
WSO.	.php	175.63 KB	2019-03-22 12:39:52	www-data/www-data	-rw-rr	RTFED
Сору	♦ submit					
		Change dir:		F	Read file:	
			submit			submit
		Make dir: [Writeabl	e]	Make fi	le: [Writeable]	
			submit			submit
		Execute:		Upload	file: [Writeable]	
			submit	Browse No files s		submit

Figure 3



2.Inconsistent Acc	ess Control
	CVSSv3 SCORE
Base	6.5
Temporal	6.5
Environmental	6.5
Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:L
	DESCRIPTION

It appears that the application does not consistently apply access controls to all its resources. The lack of access protection for some sensitive resources can be leveraged by an non-authorised attacker to either gather important information for a consequent attack against other application users, or to access and modify directly unprotected application data.

Assuming a user with a given identity, authorisation is the process of determining whether that user can access a given resource, based on the user's privileges and any permissions or other access-control specifications that apply to the resource.

When access control checks are not applied consistently - or not at all - users are able to access data or perform actions that they should not be allowed to perform. This can lead to a wide range of problems, including information exposures, denial of service, and arbitrary code execution.



ATTACK SCENARIO

The page can be identified quick and easily through application fingerprinting and crawling. An attacker could read sensitive data, either by reading the data directly from a data store that is not properly restricted, or by accessing insufficiently-protected, privileged functionality to read the data.

REMEDIATION RECOMMENDATION

This issue should be fixed by applying proper authorisation permission to the affected resources unless it is not an intended business feature.

- For web applications, make sure that the access control mechanism is enforced correctly at the server side on every page. Users should not be able to access any unauthorised functionality or information by simply requesting direct access to that page. One way to do this is to ensure that all pages containing sensitive information are not cached, and that all such pages restrict access to requests that are accompanied by an active and authenticated session token associated with a user who has the required permissions to access that page.
- Ensure that you perform access control checks related to your business logic. These checks may
 be different than the access control checks that you apply to more generic resources such as
 files, connections, processes, memory, and database records. For example, a database may
 restrict access for medical records to a specific database user, but each record might only be
 intended to be accessible to the patient and the patient's doctor.
- Reduce the attack surface by carefully mapping roles with data and functionality. Use role-based access control (RBAC) to enforce the roles at the appropriate boundaries. Note that this



approach may not protect against horizontal authorisation, i.e., it will not protect a user from attacking others with the same role.

	AFFECTED ASSET
Open	<u>bat-api.attackforge.com, bat-portal.attackforge.com</u>
NOTES	During testing, it was possible to iterate over 100k users in a short amount of time using scripts. These scripts were successful in scraping user details such as: • First name • Last name • Email address • Home address • Work address
POC	 Open a web browser in private/incognito mode Navigate to https://bat-portal.attackforge.com/api/users/1 Notice that you are able to view all user information. Increase '1' to view another user.



⑦ Choose an attack type	Start attack
Attack type: Sniper	
Payload positions	
Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.	
Target: https://gipapdiujce.shop	Add §
	Clear §
1 GET /catalog/product?productId=\$4§ HTTP/2 Host: ginandjuice.shop	Auto §
3 CGORte	Refresh
4 Sec-Ch-Ua: "Chromium";v="109", "Not_A Brand";v="99" 5 Sec-Ch-Ua-Mobile: ?0	
6 Sec-Ch-Ua-Platform: "macOS" 7 Umarda Incourse Resources 1	
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/109.0.5414.75 Safari/537.36	
<pre>9 Accept: text/html.application/xhtml+xml.application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9 10 Sec-Fetch-Site: same-origin</pre>	
11 Sec-Fetch-Mode: navigate	

Figure 4: Discover endpoint to enumerate users

Add § Clear § Auto § Refresh



Positions Payload	Resource pool	Settings
Payload sets		
You can define of each payload set	e or more payload sets and each payload type	ts. The number of payload sets depends on the attack type defined in the Positions tab. Various se can be customized in different ways.
Payload set: 1	~	Payload count: 51
Payload type: N	imbers ~	✓tequest count: 51
Payload setting	s [Numbers]	
This payload type	generates numeric pay	vyloads within a given range and in a specified format.
Number range		
Type:	🔾 Sequential 🔵 R	Random
From:	1	
To:	51	
Step:	1	
How many:		

Figure 5: Set up automation parameters



Results	Positions Payloads	s Resource pool	Settings		
Filter: Show	ving all items	\frown			
Request	Payload	Status ~	Error	Redirects followed	Timeout
13	13	404		0	
0		200		0	
1	1	200		0	
2	2	200		0	
3	3	200		0	
4	4	200		0	
5	5	200		0	
6	6	200		0	
7	7	200		0	
8	8	200		0	
9	9	200		0	
10	10	200		0	
11	11	200		0	
12	12	200		0	
1					

Figure 6: Notice 200 responses are successful



3.Relative Path Tro	versal
	CVSSv3 SCORE
Base	4.3
Temporal	4.3
Environmental	4.3
Vector	CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N
	DESCRIPTION

The software uses external input to construct a pathname that should be within a restricted directory, but it does not properly neutralize sequences such as '..' that can resolve to a location that is outside of that directory.

This allows attackers to traverse the file system to access files or directories that are outside of the restricted directory.

ATTACK SCENARIO

The attacker may be able to create or overwrite critical files that are used to execute code, such as programs or libraries.

The attacker may be able to overwrite or create critical files, such as programs, libraries, or important data. If the targeted file is used for a security mechanism, then the attacker may be able to bypass



that mechanism. For example, appending a new account at the end of a password file may allow an attacker to bypass authentication. The attacker may be able read the contents of unexpected files and expose sensitive data. If the targeted file is used for a security mechanism, then the attacker may be able to bypass that mechanism.

For example, by reading a password file, the attacker could conduct brute force password guessing attacks in order to break into an account on the system. The attacker may be able to overwrite, delete, or corrupt unexpected critical files such as programs, libraries, or important data. This may prevent the software from working at all and in the case of a protection mechanisms such as authentication, it has the potential to lockout every user of the software.





Figure 7

REMEDIATION RECOMMENDATION

Assume all input is malicious. Use an 'accept known good' input validation strategy, i.e., use a whitelist of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does.

When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. As an example of business rule logic, 'boat' may be



syntactically valid because it only contains alphanumeric characters, but it is not valid if the input is only expected to contain colours such as 'red' or 'blue.'

Do not rely exclusively on looking for malicious or malformed inputs (i.e., do not rely on a blacklist). A blacklist is likely to miss at least one undesirable input, especially if the code's environment changes. This can give attackers enough room to bypass the intended validation. However, blacklists can be useful for detecting potential attacks or determining which inputs are so malformed that they should be rejected outright. When validating filenames, use stringent whitelists that limit the character set to be used. If feasible, only allow a single '.' character in the filename to avoid weaknesses such as CWE-23, and exclude directory separators such as '/' to avoid CWE-36. Use a whitelist of allowable file extensions, which will help to avoid CWE-434. Do not rely exclusively on a filtering mechanism that removes potentially dangerous characters. This is equivalent to a blacklist, which may be incomplete (CWE-184).

For example, filtering '/' is insufficient protection if the filesystem also supports the use of " as a directory separator. Another possible error could occur when the filtering is applied in a way that still produces dangerous data (CWE-182). For example, if '../' sequences are removed from the '.../..//' string in a sequential fashion, two instances of '../' would be removed from the original string, but the remaining characters would still form the '.../' string.

AFFECTED ASSET		
Open	<u>bat-portal.attackforge.com</u>	
POC	Authenticate to the portal as administrator user and browse to the URL:	following

Slide 26 of 29



https://globexcorp.com.au/test/cmsedit.jsp?file=///////etc/hostn
ame
Note that the resulting page will contain the hostname of the underlying
system.
Affected
• URL:
https://globexcorp.com.au/test/cmsedit.jsp?file=///////etc/
hostname
Parameter: file



Go Cancel < T > T					
Request	Response				
Raw Params Headers Hex	Raw Headers Hex HTML R				
GET /webgoat.net/Content/BathWanipulation.a spx?filename /Web.config HTTP/1.1 Host: 172.16 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.11; rv:45.0) Gecko/20100101 Firefox/45.0 Accept: text/html,application/xhtml+xml,applica tion/xml;q=0.9,*/*;q=0.8 Accept-Language: en-GB,en;q=0.5 Accept-Encoding: gzip, deflate Referer: http://172.16.67.136/webgoat.net/Conten t/PathManipulation.aspx Cookie: ASP.NET_SessionId=77A596BF1DD9CB8E0423 020F; remember_token=PNkIxJ3DG81XL0F4vrAWBA;	HTTP/1.1 200 OK Date: Tue, 12 Apr 2016 03:41: Server: Apache/2.2.14 (Ubuntu mod_ssl/2.2.14 OpenSSL/0.9.8) Accept-Ranges: bytes Connection: Keep-Alive, close Content-Disposition: attachme X-AspNet-Version: 2.0.50727 Content-Length: 9265 Cache-Control: private Content-Type: application/oct xml version="1.0"? <br Web.config file for DotNetGov The settings that can be used http://www.mono-project.com/o				

Figure 8: Identified path traversal



Go Cancel < * > *					
Request	Response				
Raw Params Headers Hex	Raw Headers Hex HTML Rend				
GET /mutillidae/index.php?page=%2f%2	ion: 2.6.3.1 Security Level: 0 (
%2f%2f%2f%2f%2f%2f%2f%2	curity Reset DB View Log				
Bost: ./././././././././././././././././././	tc/passwd root:x:0:0:root:/root sync:x:4:65534:sync				
Gecko/20100101 Firefox/44.0	mail:x:8:8:mail:/var/				
text/html,application/xhtml+xml,appl	www-data:x:33:33:w				
ication/2ml;g=0.9,*/*;g=0.8 Accept-Language: en-GB,en;g=0.5	nobody:x:65534:655				
Accept-Encoding: gzip, deflate	klog:x:102:103::/hol				
http://172.16.67.136/mutillidae/	messagebus:x:107:1				
Cookle: showhints=0; username=admin; uid=1;	polkituser:x:109:118				
remember token=PNkIxJ3DG81XL0F4vr&WB	pulse:x:111:120:Puls				

Figure 9: Accessing passwd