

DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Secure Coding Phase 2

Team 3: Patrick Sattler, Aurel Roci, Stefan Kohler



Contents

1	Tim	Time Tracking Table						
2	Vul	nerabil	ities Overview	2				
	2.1	Secod	e21	2				
		2.1.1	Static Session ID	2				
		2.1.2	Stored XSS in Registration	2				
		2.1.3	Missing Lock Out Mechanism	3				
	2.2	Team3	3 Online Banking	3				
	2.3		rability Overview	3				
3	Det	ailed R	eport	4				
	3.1	Config	guration and Deploy Management Testing	5				
		3.1.1	Test File Extensions Handling for Sensitive Information(OTG-					
			CONFIG-003)	5				
		3.1.2	Test HTTP Methods(OTG-CONFIG-006)	6				
		3.1.3	Test HTTP Strict Transport Security(OTG-CONFIG-007)	7				
		3.1.4	Test RIA cross domain policy(OTG-CONFIG-008)	8				
	3.2	Identi	ty Management Testing	9				
		3.2.1	Test Role Definitions(OTG-IDENT-001)	9				
		3.2.2	Test User Registration Process(OTG-IDENT-002)	10				
		3.2.3	Test Account Provisioning Process(OTG-IDENT-003)	11				
		3.2.4	Testing for Account Enumeration and Guessable User Account(OTG-					
			IDENT-004)	12				
		3.2.5	Testing for Weak or unenforced username policy(OTG-IDENT-005)	13				
	3.3	Authe	entication Testing	14				
		3.3.1	Testing for Credentials Transported over an Encrypted Channel (OTG-	-				
			AUTHN-001)	14				
		3.3.2	Testing for default credentials(OTG-AUTHN-002)	15				
		3.3.3	Testing for Weak lock out mechanism(OTG-AUTHN-003)	16				
		3.3.4	Testing for bypassing authentication schema(OTG-AUTHN-004)	17				

Contents

	3.3.5	Test remember password functionality(OTG-AUTHN-005)	18
	3.3.6	Testing for Browser cache weakness(OTG-AUTHN-006)	19
	3.3.7	Testing for Weak password policy(OTG-AUTHN-007)	20
	3.3.8	Testing for Weak security question/answer(OTG-AUTHN-008).	21
	3.3.9	Testing for weak password change or reset functionalities(OTG-	
		AUTHN-009)	22
	3.3.10	Testing for Weaker authentication in alternative channel(OTG-	
		AUTHN-010)	23
3.4	Autho	rization Testing	24
	3.4.1	Testing Directory traversal/file include(OTG-AUTHZ-001)	24
	3.4.2	Testing for bypassing authorization schema(OTG-AUTHZ-002).	25
	3.4.3	Testing for Privilege Escalation(OTG-AUTHZ-003)	26
	3.4.4	Testing for Insecure Direct Object References(OTG-AUTHZ-004)	27
3.5	Session	n Management Testing	28
	3.5.1	Testing for Bypassing Session Management Schema(OTG-SESS-001)	28
	3.5.2	Testing for Cookies attributes(OTG-SESS-002)	29
	3.5.3	Testing for Session Fixation(OTG-SESS-003)	30
	3.5.4	Testing for Exposed Session Variables(OTG-SESS-004)	31
	3.5.5	Testing for Cross Site Request Forgery(OTG-SESS-005)	32
	3.5.6	Testing for logout functionality(OTG-SESS-006)	33
	3.5.7	Test Session Timeout(OTG-SESS-007)	34
	3.5.8	Testing for Session puzzling(OTG-SESS-008)	35
3.6	Data \	Validation Testing	36
	3.6.1	Testing for Reflected Cross Site Scripting(OTG-INPVAL-001)	36
	3.6.2	Testing for Stored Cross Site Scripting(OTG-INPVAL-002)	37
	3.6.3	Testing for HTTP Verb Tampering(OTG-INPVAL-003)	38
	3.6.4	Testing for HTTP Parameter pollution(OTG-INPVAL-004)	39
	3.6.5	Testing for SQL Injection (OTG-INPVAL-005)	40
	3.6.6	Testing for XML Injection(OTG-INPVAL-008)	41
	3.6.7	Testing for SSI Injection(OTG-INPVAL-009)	42
	3.6.8	Testing for XPath Injection(OTG-INPVAL-010)	43
	3.6.9	IMAP/SMTP Injection(OTG-INPVAL-011)	44
	3.6.10	Testing for Code Injection, Testing for Local File Inclusion, Testing	
		for Remote File Inclusion(OTG-INPVAL-012)	45
	3.6.11	Testing for Command Injection(OTG-INPVAL-013)	46

Contents

	3.6.12	Testing for Buffer overflow, Testing for Heap overflow, Testing for	
		Stack overflow, Testing for Format string (OTG-INPVAL-014)	47
	3.6.13	Testing for incubated vulnerabilities(OTG-INPVAL-015)	48
	3.6.14	Testing for HTTP Splitting/Smuggling(OTG-INPVAL-016)	49
3.7	Error 1	Handling	49
3.8	Crypto	ography	49
3.9	Busine	ess Logic Testing	50
	3.9.1	Test Business Logic Data Validation(OTG-BUSLOGIC-001)	50
	3.9.2	Test Ability to Forge Requests(OTG-BUSLOGIC-002)	51
	3.9.3	Test Integrity Checks(OTG-BUSLOGIC-003)	52
	3.9.4	Test for Process Timing(OTG-BUSLOGIC-004)	53
	3.9.5	Test Number of Times a Function Can be Used Limits(OTG-	
		BUSLOGIC-005)	54
	3.9.6	Testing for the Circumvention of Work Flows(OTG-BUSLOGIC-006)	55
	3.9.7	Test Defenses Against Application Mis-use(OTG-BUSLOGIC-007)	56
	3.9.8	Test Upload of Unexpected File Types(OTG-BUSLOGIC-008)	57
	3.9.9	Test Upload of Malicious Files(OTG-BUSLOGIC-009)	58
3.10	Client	Side Testing	58

1 Time Tracking Table

2 Vulnerabilities Overview

Based on our testing, we identified the following vulnerabilities for the Secode21 Bank and the OnlineBanking Bank:

2.1 Secode21

2.1.1 Static Session ID

Likelihood: *high*Implication: *high*

•Risk: *high*

•Reference: OWASP OTG-SESS-003 (see section ??)

The session id is saved in form of the (static) user id in a cookie. This cookie can be used on any machine to take over the account of a user. The lifetime of this cookie is only limited by the cookie lifetime field.

2.1.2 Stored XSS in Registration

Likelihood: *medium*Implication: *high*

•Risk: *high*

•Reference: OWASP OTG-INPVAL-002 (see section ??)

Using stored cross-site-scripting attacks, one can inject JavaScript code, that is run, when the Administrator/Employee logs in. Arbitrary code can be loaded from a third party page.

2.1.3 Missing Lock Out Mechanism

Likelihood: *high*Implication: *medium*

•Risk: medium

•Reference: OWASP OTG-AUTHN-003 (see section ??)

The application has no lock out mechanism, which allows brute force attacks on known usernames and testing for a valid password

2.2 Team3 Online Banking

2.3 Vulnerability Overview

3 Detailed Report

The following pages describe for each test how both applications Secode21 and Online Banking Bank performed. The test is divided in different sections following the OWASP Testing Guide v4.

3.1 Configuration and Deploy Management Testing

3.1.1 Test File Extensions Handling for Sensitive Information(OTG-CONFIG-003)

3.1.2 Test HTTP Methods(OTG-CONFIG-006)



3.1.4	Test RIA	cross domain	policy(OTG-	CONFIG-008)
-------	----------	--------------	-------------	-------------

3.2 Identity Management Testing

3.2.1 Test Role Definitions(OTG-IDENT-001)

3.2.2	Test	User	Registi	ration	Process(OTG	-IDE	NT-00	02)

3.2.3	Test Acco	ount Prov	visioning	Process(O	TG.	-IDENT-00)3)
-------	-----------	-----------	-----------	-----------	-----	-----------	-----

3.2.4 Testing for Account Enumeration and Guessable User Account(OTG-IDENT-004)



3.3 Authentication Testing

3.3.1 Testing for Credentials Transported over an Encrypted Channel(OTG-AUTHN-001)

3.3.2	Testing	for	defaul	t cred	lential	s(O)	G-AU	JTHN	I-002)
-------	---------	-----	--------	--------	---------	------	------	------	--------

2	Deta	المداني	D aa	ant
	1 1210	mea	KPn	ori

3.3.3	3.3.3 Testing for Weak lock out mechanism(OTG-AUTHN-003)						

3.3.4	Testing for bypassing authentication schema(OTG-AUTHN-004)

2	Deta	•1 1	ח	
۲.	1 10t/	าาเอก	K on	αvt

3.3.5	Test remem	ber password	functionality	7(O	TG-AUTHN	[- 005)
-------	------------	--------------	---------------	------------	----------	-----------------

3.3.6	Testing 1	for	Browser	cache	weal	kness((\mathbf{O})	ΓG-A	4U	THN	I-006)
-------	-----------	-----	---------	-------	------	--------	----------------	------	----	-----	--------------	---



3.3.8	Testing for Weak security question/answer(OTG-AUTHN-008)	

3.3.9 Testing for weak password change or reset functionalities (OTG-AUTHN-009)

3.3.10 Testing for Weaker authentication in alternative channel (OTG-AUTHN-010)

3.4 Authorization Testing

3.4.1 Testing Directory traversal/file include(OTG-AUTHZ-001)

$^{\circ}$	Detailed	D
1	1 10†01100	K onort

3.4.2	Testing for bypassing authorization schema(OTG-AUTHZ-002)

3.4.3	Testing	for	Privileg	e Esca	lation((OT	G-AL	J THZ-	003)
-------	----------------	-----	----------	--------	---------	-----	------	---------------	------

3.4.4	Testing for Insecure Direct Object References(OTG-AUTHZ-004)	

3.5 Session Management Testing

3.5.1 Testing for Bypassing Session Management Schema(OTG-SESS-001)

3 . 5.2 7	Testing	for	Cookies	attributes(OT	G-SESS	5-002)
------------------	----------------	-----	---------	-------------	----	--------	--------

3.5.3 Testing for Session Fixation(OTG-SESS-003)

Secode21 Likelihood: 0
Impact: 0
Risk:0

			HISINO
		Secode21	
Observation	N/A		
Discovery	N/A		
Likelihood	N/A		
Implication	N/A		
Recommendations	N/A		

Metric	Value
Access Vector	
Attack Complexity	
Privileges Required	
User Interaction	
Scope	
Confidentiality Impact	
Integrity Impact	
Availability Impact	

3 Detailed Report

3.5.4	Testing	for Ex	posed	Session	Variables	(OTG-SESS-004	.)
-------	----------------	--------	-------	---------	------------------	---------------	----

	3	Detailed	Report
--	---	----------	--------

3.5.5	Testing f	or Cross	Site Request	Forgery(O	TG-SESS-005)
-------	-----------	----------	--------------	-----------	--------------

3.5.6	Testing for	logout	functionalit	y(OTG	SESS-006)
		0		·	,

3.5.7	Test S	Session	Timeout(OT	G-9	SESS-	007
J.J.1	TCSL C	JESSIUII	IIIIICUUU	$\mathbf{O}_{\mathbf{I}}$	O-1	JEOU	.007

3.5.8	Testing f	for Session	puzzling(OT	G-SESS-008)

3.6 Data Validation Testing

3.6.1 Testing for Reflected Cross Site Scripting(OTG-INPVAL-001)

	Likelihood: 0	
Secode21	Impact: 0	
	Risk:0	
	Secode21	
Observation	We observed no reflected cross site scripting vulnerability.	
Discovery	It seems that all parameters are stored in the database before inserting the	
	values in the HTML	
Likelihood	N/A	
Implication	N/A	
Recommendations	N/A	

Metric	Value
Access Vector	
Attack Complexity	
Privileges Required	
User Interaction	
Scope	
Confidentiality Impact	
Integrity Impact	
Availability Impact	

3.6.2 Testing for Stored Cross Site Scripting(OTG-INPVAL-002)

	Likelinood: /
Secode21	Impact: 10
	Risk: 10

	Risk: 10
	Secode21
Observation	We observed several possibilities to execute a stored XSS attack. But
	not all of them could be exploited as the length of the corresponding
	database fields was often very restricted. We manually tried to inject
	JavaScript code in every input field. Therefore we used the following
	code, which just alerts a message.
Discovery	We inserted Javascript code in the name field on the register page.
	When we logged in as an employee the script was executed. There were
	cases when the script caused for new registered users after the script
	was entered to not appear.
Likelihood	This vulnerability can be easily detected, but require some JavaScript
	knowledge to exploit it. But the BeEF framework allows to quickly test
	several attacks, therefore we estimated the likelihood to be medium.
Implication	The implications are severe as we proofed that it is possible to steal
	the session. As we injected the code on the admin landingpage, which
	implies that we were able to act as an admin and register an abitrary
	account.
Recommendations	
	Try to use whitelisting for the different datatypes and do not rely on
	the frontend input validation.

Metric	Value
Access Vector	
Attack Complexity	
Privileges Required	
User Interaction	
Scope	
Confidentiality Impact	
Integrity Impact	
Availability Impact	

3.6.3 Testing for HTTP Verb Tampering(OTG-INPVAL-003)

Likelihood: 0 Secode21 Impact: 0 Risk: 0 Secode21 Observation We did not observe any notable behaviour. We used the Zed Attack Proxy (ZAP) to change the HTTP requests method Discovery to the ones listed below. The requests that were allowed responded with the index page or an empty body. The rejected requests responded with an error message in the body Methods that were allowed • HEAD • OPTIONS • GET • POST • PUT Methods that were rejected • TRACE • CONNECT N/A Likelihood Implication N/A N/A Recommendations

Metric	Value
Access Vector	
Attack Complexity	
Privileges Required	
User Interaction	
Scope	
Confidentiality Impact	
Integrity Impact	
Availability Impact	

3	Detailed	Report

3.6.4 Testing for HTTP Parameter po	ollution(OTG-INPVAL-004)
-------------------------------------	--------------------------

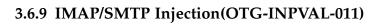
3.6.5	Testing	for SQ	L Injection	ı (OTG-IN)	PVAL-005)
-------	----------------	--------	-------------	------------	-----------

3.6.6 Testing for XML Injection(OTG-INPVAL-008)	3.6.6	Testing	for	XML	Inj	ection	(\mathbf{O}^{r})	ΓG	-INP	VAI	L-008)
---	-------	----------------	-----	-----	-----	--------	--------------------	----	------	-----	--------------	---

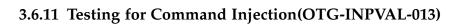
3.6.7 Testing for SSI Injection(OTG-INPVAL-009	3.6.7	Testing for	SSI In	jection(OTG	-INPVAL-009
--	-------	--------------------	--------	-------------	-------------

~	\mathbf{D}	ailed	ח	
~	1 101	ппоп	K on	αv_t
.,	DUU	ииси	$-1 \times U$	UIL

3.6.8 Testing	for	XPath	In	iection((OΤ	G-I	NPV	/AL-(010)
---------------	-----	-------	----	-----------	----	-----	-----	-------	------



3.6.10 Testing for Code Injection, Testing for Local File Inclusion, Testing for Remote File Inclusion(OTG-INPVAL-012)



3.6.12 Testing for Buffer overflow, Testing for Heap overflow, Testing for Stack overflow, Testing for Format string (OTG-INPVAL-014)

2	Deta	الممالنة	D an	~~4
	1 1211	тичи	KPD	m

3.6.13	lesting for incubated vulnerabilities(O1G-INPVAL-015)

3.6.14 Testing for HTTP Splitting/Smuggling(OTG-INPVAL-016)

3.7 Error Handling

Secode21

Secode21 does not provide a lot of error messages for incorrect inputs (e.g. incorrect TAN length, wrong TAN, TAN used).

Based on the client side input validation, there are also no messages for manipulated input via proxy or by removing the validation patterns, which can lead to problems. Examples would be a malformated email which results in a not working account or a longer input then expected, which cuts off the end of the input. There are some cases when the page returns the path of the file where the error occurred.

Team3 Online Banking

3.8 Cryptography

3.9 Business Logic Testing

3.9.1 Test Business Logic Data Validation(OTG-BUSLOGIC-001)

3.9.2 Test Ability to Forge Requests(OTG-BUSLOGIC-00	3.9.2	Test Abilit	y to Forge	Requests	(OTG	-BUSLC)GIC-002
--	-------	-------------	------------	----------	------	--------	----------

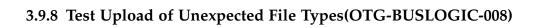


3.9.4	Test for	Process	Timing(OTC	G-BUSL	OGIC-004)
-------	----------	----------------	---------	-----	--------	-----------

3.9.5 Test Number of Times a Function Can be Used Limits(OTG-BUSLOGIC-005)







3.9.9 Test Upload of Malicious Files(OTG-BUSLOGIC-009)

3.10 Client Side Testing