	Java Script Vulnerabilities
	The state of the s
-X	Exploiting Javascript Vulnerabilities Can manipulate
	Data, reducent Sessions, modify and Stear data, and much more.
7.19	More.
	Although Javascopt is typically thought of as
	a Client-Side Application
1	Java Script Secrify Issues Can Create problem
-	on Server-side environments as well
1	The good more way carlotte your contine was the
المذاللة	Could proper materials at in the Les
	+ What are Common Java Script Volnerabilities?
	1 most Common Javasupt affacts vectors molude
	-> execting malicious scripts,
	-) stealing a user's established session data (or)
	data from browser's Local Storage.
	-> too tricking users into performing unintended achains.
	and exploining vulnerabilities in Source code of
N	web Application
	* Common Javascript Seonly Vulnevabilities
100	Cross-sike Scripting (xss)
	Cross-sife request forgery (CSRF)
	Third-party Seenty Velaurableics
	SQL Trisection
	Senstre Cookie exposure
	La sullive de la company de la

	Cross-site Serping: (XSS):
	is one of the most andespread Seenty not
	in web applications.
	# It occoon when an affactor injects malicious
,	Code into the closest and of the application.
	* This normally happens when an application
	accepts unhusted (or over-Suppled) data on
•	1 1 100 With and Brown Or Waltdohe
	a webpage without Escapag or valedang
- 7/4	if properly
	S. A. Sunday Control of the Man de la control of the control of th
	-> If an affaction exploits vulnerabilities, they
	Could perform maliances achous like tampering,
	data theft, remote Conhol, or even malwave
	deshibution
	and the state of t
-	to prevent XSS attacks, developers Should separate
	to prevent XSS attacks, developers Should separate unhuned data or userinput from achive browser
	1 onker.
Hair	In javascript you can achen this by
	· Validating and Sanitizing mout from users to
	ensure It only Contains acceptable Charecers that
	Can not be used to launch & ss attacks
	Carle Marie Contraction of the C
	· 11 chia Pala mellada Suel as man fort for
	· Using Safe methods Such as inner text for
	Manipulating Dom
	dangerous consent, then by prenenting Dom-boxed xss
	Congress Content, the by poening com-was ys
	W Mais Example more land accordingly
	Voluerabilities by design
	voucerasius by almogn

	In the example above, the application gets user Ins from URLS and refung values the Corresponding email address by genery the defaberse.
	User In's from URL'S and refung refuses the
	Corresponding email address by geoneug the
	dafabase.
	Two things are living in the code Snipper
	First thing for db query is built using a String Concepenation.
in	Concefenation.
	The second issue is flat the user input
-	1's Compected to the grey Mead of being
	The second issue as flat the user input 1's Comnected to the grey islead of being hardled as unpuryted data
	The second of th
	C. Steinman & Commencer Co
	To mitigate SQL injections, developers Should always perform proper upst validation. Swhen input from the user fails the Validation
	always perform proper most validation.
	-> when most from the user fails the volidarian
	Checks, the SOL guory is not executed,
	-> Another way of preventing SQL Injection is using parametrized gones (on) prepared statements
t end t	parametrized gones (or) prepared statements
	Inpend of Concomunations
	the state of the s
	Example:
	// Wrong implementation
	11 unvalidated 'chan Nam' Bravameter amade
marata transpa	to the grony allows an attacker to inject any sp
	Code,
4	
	Shug gray = "SELECT account balance from
	The Only Allege was a
	USA-Data WHERE USEX-MAME = " + request-get Pavameter ["Cleant Non
	1 cquest get continuer. Chart Nan

how to use xs-fillers
DECTOS SEED LIGHT CONTROLLED SEED SEED SEED SEED SEED SEED SEED
Example & In well of them I would be sugar
Const express = require ('express'):
Const xsspillers = require ('xss-fillers');
Const Util = require ('Util');
const app = express c:
app. get ('1', (reg, res) =) & Const Usafe psim Name =
reg. quen fine vaine;
Const Sofe Girl Name =
Xss Fillers. in HTMl Data (unsafe Frymaine):
res. Send (Upl. format ('Lw> Tomys < 1h15,
SafeFixyName); 3);
in miliging the inferious policing per should
descript and in the finder and the delicition for
SOL Injection:
· Clocks the set and it wis of the
SQL database are vulnerable to injection affaces
Where guery parameters are exploited to execute
arbifarary Inspection
C 1 2 A A 1 2 A C l'anni a 1 1 3 C
Const express = required ('express'):
Const db = require (! 1616');
0.00
Const Youter = Cxpien Louger():
voujer.get ('lemail', (rea, res) => }
db. greny ('SBIECT CMall from Wsers where
id = + req. quy. id):
· then ((record) => §
1110gral frow
res. Send (record EoD);
The same of the sa
3);

In the example above, the application gets
User ID'S from URL'S and refused vetures the
Corresponding email address by groneing the Two flugs are living in the code Snippet FIRST thing for db query is built vong a String Concofenapas The second issue is that the user most I's Compected to the grey Islead of boug hardled as conjunted data, To mitigate SQL injections, developers Should always perform proper upst validation.

-> when input from the user fails the validation Checks, the SOL guory is not -> Another way of preventing SQL Injection is Using parametrized quisos (or) prepared statements

Imend of Concernations Example: 11 wrong implementation 11 unvalidated "cheat Nam" pravameter approduce to the grony allows an attacker to inject any spe Shing gray = "SELECT account balance from USA-Date WHERE USEN MAME = + request get Pavameter ["Cleans Non

Statement = Connection ConfStatement Resultset result = Statement exectepolery (query); Morred implementation. regress, getparameter (" cheut Name"): validate input to detect attacks. String query "SELECT account-balance FROM user_date WHERE User-name = ?"; Prepared Statement pamt = Connection. prepared Statement (query); PStmt. setSting (1; Clientname): Kesulfet results = po pstmt. Exerce Query (); Sensitive Cookie exposure: The Cheat side Script on Every browser Can access all the confert referred by applianon to the Server -> This includes cookies that offen Confain Seulihur data Such as Session Ini -> Exposing Session identifiers, exeafler in Urls, Error messages, or longs is a bad prache the opens up an applianou to Seeinty issues like

	Ind Servician Fixation > 18 prevent this, developen Must Consistently Use Hospis and implement Hospiscopers. > The Hospiscoper affiners in Cookies tells the browser to prevent cookie access through the DOM. > By Doing this Clent-side Script affacts are prevented from accessing Sensitive class stered in Cookies > Another way of Sening user Session 'y oping by par-requests as apposed to vong par-session Identifiers.
	Se Hotps and implement title only contess The Hotp-only affilled in Cookies tells the browser to prevent cookie access through the DOM: By poing this cheut side Script affacts are prevented from accessing Sensitive data stored in Cookies Another way of Jenny user Session 'y oping by par-requests as apposed to vary par-session
	Just Hap-only affillede In Cookies this fee browser to prevent cookies through the DOM. The point this cheek side Script affacts are prevented from accessing Scription data stend in Cookies The prevented from accessing Scription data stend in Cookies The prevented from accessing Scription data stend in Cookies The Hap-only affilbete In Cookies this fee of the Script access five access five access five access for stend in Cookies The Hap-only affilbete In Cookies this fee of the Server access five access through the prevented from accessing Scription access through the prevented from accessing Scription access five acc
	DOM: The poing this chem side Script affacts are Prevented from accessing Sensitive class stored in Cookies The parties of Some user Session 'y oping by par-requests as opposed to using por-session
	Tople. The poing this chew side Script affacts are prevented from accessing Scriptive data stoned in Cockies The Another way of Seeing user Session 'y opting by par-requests as opposed to very par-session
	The poing this chem side Script affacts are prevented from accessing Sensitive data stered in Cockies The Another way of Seeing User Session 'y opting by par-requests as opposed to using par-session
	-> Another way of Seeing user Sessions 'y opting by par-requests as apposed to using par-session
	-> Another way of Seeing user Sessions 'y opting by par-requests as apposed to using par-session
	-> Another way of Seeing user Sessions 'y opting by par-requests as apposed to very por-session Identifiers.
	-> Another way of Second User Session 4 opting by par-requests as opposed to using per-session Identifiers.
	by par-requests as opposed to using par-session. Identifien.
	Identifien.
	the sale of the sa
	(J. K. J. C.
1	Dufing an fogether:
	Adopha good Modes Prophes
	Can secure applications against Common Javascryt
	Volnerabilities on both the Clour-Side and
	그 이 사람이 있다는 그 회문에는 그 이 사람들은 그 이 사람들이 되었다면 하지만
	generales Com Cont Paris le la Cudata
	When using Java Sarpt follow the Key Gudeky
	Never trust user input Encrypt data transmitted
	Use proper encoding les coping between the chart of Son Son Son Use Secure Components and
1	Defin a Confert Secrety policy Aps Set Secure Cookies