

Forum:[Hack Forums](#)

Thread: [Starting out Pentesting](#)

Th4nat0s - March 19, 2017, 7:21 am

Hello guys,

so I got interested in pentesting (web-application precisely) after I found out about hackerone, I searched the web about good tutorials and such but they were all badly structured, I don't really know where to start and what are the basics.

I hope you can guide me.

Thanks

KalibRx - March 19, 2017, 9:26 am

Good basic tutorials: <https://null-byte.wonderhowto.com/>

2 Good starter books for python (writing mini exploits):

http://multimirrorupload.com/ss_h2qdzz37rmtm and http://multimirrorupload.com/km806n0p3kjs/a_whirlwind_tour_of_python.pdf

And you'll need an understanding of *NIX based systems:

http://multimirrorupload.com/so_8othcl77bs2v

zk34911 - March 19, 2017, 9:44 am

^

He is interested in Web Application Exploitation, can't you fucking read?

WTF are you even suggesting him? Python directly to automate his stuff in networking and all when he doesn't even know basic shit?

WHY?

Give me a simple reason why a person should start learning Python instead of CSS, HTML, PHP, SQL, JavaScript & HTTP headers if he wants to learn Web Application Exploitation

WTF is wrong with this community

Th4nat0s - March 19, 2017, 10:00 am

(Mar 19, 2017 09:44 AM)zk34911 Wrote: [\[CITE\]](#) ^

He is interested in Web Application Exploitation, can't you fucking read?

WTF are you even suggesting him? Python directly to automate his stuff in networking and all when he doesn't even know basic shit?

WHY?

Give me a simple reason why a person should start learning Python instead of CSS, HTML, PHP, SQL, JavaScript & HTTP headers if he wants to learn Web Application Exploitation

WTF is wrong with this community

So I should start with learning CSS, HTML, JavaScript etc ... and HTTP.

After that how did you make the transition to finding exploits, is there any good books or ressources about that ?

KalibRx - March 19, 2017, 10:13 am

(Mar 19, 2017 09:44 AM)zk34911 Wrote: [\[CITE\]](#) ^

He is interested in Web Application Exploitation, can't you fucking read?

WTF are you even suggesting him? Python directly to automate his stuff in networking and all when he doesn't even know basic shit?

WHY?

Give me a simple reason why a person should start learning Python instead of CSS, HTML, PHP, SQL, JavaScript & HTTP headers if he wants to learn Web Application Exploitation

WTF is wrong with this community

You're obviously a child. Why would anyone that has no interest in development, rather only (at least specified) learn anything about CSS? Let me break it down for you little boy:

Reading REAL articles about security, and penetration testing will give code examples. ALWAYS. Do yourself a favor and look at exploit-db. See a pattern? THAT'S RIGHT! All of them have code examples! So is he going to learn HOW to do things, or WHY things work the way they do? If you want to be spoon fed, sure, don't learn python. Apparently you know NOTHING of the sort if you're even questioning FOR A SECOND the validity of learning python with regard to pen-testing.

I beg that you do not listen to zk34911. He is a frightfully unseemly delinquent and a simple-minded cesspool of sub-human filth. He is a conspicuously boorish imbecile and a monstrous air-polluting deplorable calamity of birth. He knows nothing of security, nor much else for that matter.

And another thing I just noticed that made me chuckle. You tossed HTTP in with HTML/CSS/JS. HA! You really know nothing. Lets show you something:

HTML = Hyper-Text Markup Language = (NOT PROGRAMMING).

CSS = Cascading Style Sheets = NOT PROGRAMMING.

PYTHON = PROGRAMMING.

JAVASCRIPT = SCRIPTING.

HTTP (Hyper-Text Transfer Protocol)= PROTOCOL.

You don't automate pentests with javascript you moron. You do it with python (mostly). Obviously you can use another language, but that's what you'll find the most examples on.

Xphile - March 19, 2017, 10:14 am

I don't claim to be great at web pentesting as I'm still learning that myself, here's a few resources I've been using recently:

Cybrary's Web Penetration Testing Course - (cybrary.it/course/web-application-pen-testing/)
Cybrary's Ethical Hacking Course (Modules 12, 13 & 14) - (cybrary.it/course/ethical-hacking/)
Cybrary's Advanced Pentesting Course (Module 11) - (cybrary.it/course/advanced-penetration-testing/)

As far as programming languages go, you should learn PHP and JavaScript. You should also become familiar with basic Linux operating systems (How the filesystem works, services like Apache and Nginx). Seen as many servers are running on Debian or Ubuntu. Keep in mind that there are still plenty of Windows and BSD servers present as well.

Translating this over to exploit development will become much easier after you understand the core functions of the web application in-use.

You may want to try out CTF challenges on vulnhub.com, I find them to be very fun ways of testing various hacking methods in a narrative-driven challenge that almost always involves some form of web exploitation.

[EDIT]

Forgot to mention common protocols that can sometimes be exploited:

- + HTTP(S)
- + FTP
- + SSH
- + SQL

I highly suggest learning the basics of networking before digging too deep. Learning how computers interact with one-another is critical when auditing anything.

[/EDIT]

Hope this helped. ?

KalibRx - March 19, 2017, 10:19 am

For the love of god. SQL is not a programming language. It's a querying language. Big difference. How does anyone learn anything here with all this misinformation spread. Take a class dude. Learn from people who know, not people who read one article and think they're "1337".

Xphile - March 19, 2017, 10:22 am

(Mar 19, 2017 10:19 AM)ByteKnight Wrote: [\[CITE\]](#) For the love of god. SQL is not a programming language. It's a querying language. Big difference. How does anyone learn anything here with all this misinformation spread. Take a class dude. Learn from people who know, not people who read one article and think they're "1337".

Apologies, I didn't intend for it to be seen as another programming language. Will edit my original post.

Th4nat0s - March 19, 2017, 10:27 am

Thanks for the suggestions guys :)

KalibRx - March 19, 2017, 10:38 am

(Mar 19, 2017 10:22 AM)D3M0GORG0N Wrote: [\[CITE\]](#) Apologies, I didn't intend for it to be seen as another programming language. Will edit my original post.

Sorry man. I honestly thought you were the guy above that I rekt.

zk34911 - March 19, 2017, 10:38 am

(Mar 19, 2017 10:13 AM)ByteKnight Wrote: [\[CITE\]](#) You're obviously a child. Why would anyone that has no interest in development, rather only (at least specified) learn anything about CSS? Let me break it down for you little boy:

Reading REAL articles about security, and penetration testing will give code examples. ALWAYS. Do yourself a favor and look at exploit-db. See a pattern? THAT'S RIGHT! All of them have code examples! So is he going to learn HOW to do things, or WHY things work the way they do? If you want to be spoon fed, sure, don't learn python. Apparently you know NOTHING of the sort if you're

even questioning FOR A SECOND the validity of learning python with regard to pen-testing.

I beg that you do not listen to zk34911. He is a frightfully unseemly delinquent and a simple-minded cesspool of sub-human filth. He is a conspicuously boorish imbecile and a monstrous air-polluting deplorable calamity of birth. He knows nothing of security, nor much else for that matter.

And another thing I just noticed that made me chuckle. You tossed HTTP in with HTML/CSS/JS. HA! You really know nothing. Lets show you something:

HTML = Hyper-Text Markup Language = (NOT PROGRAMMING).

CSS = Cascading Style Sheets = NOT PROGRAMMING.

PYTHON = PROGRAMMING.

JAVASCRIPT = SCRIPTING.

HTTP (Hyper-Text Transfer Protocol)= PROTOCOL.

You don't automate pentests with javascript you moron. You do it with python (mostly). Obviously you can use another language, but that's what you'll find the most examples on.

"You're obviously a child. Why would anyone that has no interest in development, rather only (at least specified) learn anything about CSS? Let me break it down for you little boy"

>> I am 19, thanks for your concern. I didn't said you need to become a programmer when I stressed on the language part, I stressed enough on the fact that you at least need to have a pre-requisite knowledge of HTML and CSS. You know why? Because IT FUCKING HELPS and even when you transition to learn PHP and such, doesn't necessarily mean it would help you in penetrating testing and finding vulnerabilities within. LEARN TO READ. Can you see who is a child now?

"Reading REAL articles about security, and penetration testing will give code examples. ALWAYS. Do yourself a favor and look at exploit-db. See a pattern? THAT'S RIGHT! All of them have code examples! So is he going to learn HOW to do things, or WHY things work the way they do?"

>> Yes, they will. I know exploit-db, thank you. You can't stress on the fucking fact that learning via code examples and patterns will remove the importance of learning programming languages down from the basics. YOU ARE FUCKING RETARDED. He is not going to learn how things work or what the fuck is happening if he doesn't know shit or basics. There has to be pre-requisite you moron. WTF IS WRONG WITH THIS FORUM

"If you want to be spoon fed, sure, don't learn python. Apparently you know NOTHING of the sort if you're even questioning FOR A SECOND the validity of learning python with regard to pen-testing."

>> I am not questioning the validity of python with regards to its importance and pen-testing. I am questioning the validity of your opinion regarding to his situation. Don't be pissed, shit happens.

"I beg that you do not listen to zk34911. He knows nothing of security, nor much else for that matter"

>>I know nothing about Security? Seriously BRO? Check my blog, check my bug bounties. Check my Twitter/Instagram, my Bug bounty profiles. My last 3 bugs alone have been valued \$1260 from Twitter security engineering team and I have submitted 3 more so you can go fuck yourself. It's unbelievable how this community can't shut its gigantic ego for one second to shut the fuck up and listen

"He is a frightfully unseemly delinquent and a simple-minded cesspool of sub-human filth. He is a conspicuously boorish imbecile and a monstrous air-polluting deplorable calamity of birth"

>> Sorry, can't hear you over the sound of how awesome I am.

" And another thing I just noticed that made me chuckle. You tossed HTTP in with HTML/CSS/JS. HA! You really know nothing. Lets show you something"

>> Yes I put it there. I didn't associate it with any programming language or anything, show me where I said it? You fucking idiot, it's fucking common sense that it isn't a programming languages rather a protocol Don't try so hard. How old are you ? 9? 11?

"HTML = Hyper-Text Markup Language = (NOT PROGRAMMING).

CSS = Cascading Style Sheets = NOT PROGRAMMING.

PYTHON = SCRIPTING.

JAVASCRIPT = PROGRAMMING.

HTTP (Hyper-Text Transfer Protocol)= PROTOCOL."

>> I already know. Thanks for trying too hard. I thought you had a fucking common sense and could get what the fuck I meant. You are not smart. and where did I linked all these under programming language? POINT IT OUT. Also can you crawl under a rock?

" You don't automate pentests with javascript you moron. You do it with python (mostly). Obviously you can use another language, but that's what you'll find the most examples on. "

>> WHO THE FUCK TALKED about automating when you can't even do the shit FUCKING manually? WHEN YOU DONT EVEN KNOW JavaScript? LOOK IN WHAT CONTEXT I AM SPEAKING IN REGARDS TO HIS FUCKING CONDITION?

"For the love of god. SQL is not a programming language. It's a querying language. Big difference. How does anyone learn anything here with all this misinformation spread.

>> I know, We know, everybody knows this. THIS ISN'T MISINFORMATION. THIS IS COMMON SENSE. And where did I put it under programming languages? Point it out.

I've had enough of this shit in this forum

Th4nat0s - March 19, 2017, 10:45 am

(Mar 19, 2017 10:38 AM)zk34911 Wrote: [\[CITE\]](#) "You're obviously a child. Why would anyone that has no interest in development, rather only (at least specified) learn anything about CSS? Let me

break it down for you little boy"

>> I am 19, thanks for your concern. I didn't said you need to become a programmer when I stressed on the language part, I stressed enough on the fact that you at least need to have a pre-requisite knowledge of HTML and CSS. You know why? Because IT FUCKING HELPS and even when you transition to learn PHP and such, doesn't necessarily mean it would help you in penetrating testing and finding vulnerabilities within. LEARN TO READ. Can you see who is a child now?

"Reading REAL articles about security, and penetration testing will give code examples. ALWAYS. Do yourself a favor and look at exploit-db. See a pattern? THAT'S RIGHT! All of them have code examples! So is he going to learn HOW to do things, or WHY things work the way they do?

>> Yes, they will. I know exploit-db, thank you. You can't stress on the fucking fact that learning via code examples and patterns will remove the importance of learning programming languages down from the basics. YOU ARE FUCKING RETARDED. He is not going to learn how things work or what the fuck is happening if he doesn't know shit or basics. There has to be pre-requisite you moron. WTF IS WRONG WITH THIS FORUM

"If you want to be spoon fed, sure, don't learn python. Apparently you know NOTHING of the sort if you're even questioning FOR A SECOND the validity of learning python with regard to pen-testing."

>> I am not questioning the validity of python with regards to its importance and pen-testing. I am questioning the validity of your opinion regarding to his situation. Don't be pissed, shit happens.

"I beg that you do not listen to zk34911. He knows nothing of security, nor much else for that matter"

>>I know nothing about Security? Seriously BRO? Check my blog, it's in my profile.I hate how this community can't shut their gigantic ego's for one second to shut the fuck up and listen.

"He is a frightfully unseemly delinquent and a simple-minded cesspool of sub-human filth. He is a conspicuously boorish imbecile and a monstrous air-polluting deplorable calamity of birth"

>> Sorry, can't hear you over the sound of how awesome I am.

" And another thing I just noticed that made me chuckle. You tossed HTTP in with HTML/CSS/JS. HA! You really know nothing. Lets show you something"

>> Yes I put it there. I didn't associate with it you fucking idiot, it's fucking common sense that it isn't a programming languages. Don't try to be too hard. How old are you ? 9? 11?

"HTML = Hyper-Text Markup Language = (NOT PROGRAMMING).

CSS = Cascading Style Sheets = NOT PROGRAMMING.

PYTHON = SCRIPTING.

JAVASCRIPT = PROGRAMMING.

HTTP (Hyper-Text Transfer Protocol)= PROTOCOL."

>> I already know. Thanks for trying too hard. I thought you had a fucking common sense and could get what the fuck I meant. You are not smart. Also can you crawl under a rock?

" You don't automate pentests with javascript you moron. You do it with python (mostly). Obviously you can use another language, but that's what you'll find the most examples on. "

>> WHO THE FUCK TALKED about automating when you can't even do the shit FUCKING manually? LOOK IN WHAT CONTEXT I AM SPEAKING IN REGARDS TO HIS FUCKING CONDITION?

"For the love of god. SQL is not a programming language. It's a querying language. Big difference. How does anyone learn anything here with all this misinformation spread.

>> I know, We know, everybody knows this. THIS ISN'T MISINFORMATION. THIS IS COMMON SENSE. And where did I put it in programming languages? Point it out.

I've had enough of this shit in this forum

Hey man, it was a just a misunderstanding, both of you have valid arguments don't get so mad about it both of you.

Shoe - March 19, 2017, 10:46 am

(Mar 19, 2017 10:38 AM)zk34911 Wrote: [\[CITE\]](#) "I beg that you do not listen to zk34911. He knows nothing of security, nor much else for that matter"

>>I know nothing about Security? Seriously BRO? Check my blog, it's in my profile.I hate how this community can't shut their gigantic ego's for one second to shut the fuck up and listen.

I'd recommend taking some time to tidy up your blog and publish some content other than a brief portfolio. From a brief look at the page, I didn't believe you had any credentials what-so-ever to back up your bounty claims, but Twitter supplemented that. I'd recommend hyperlinking to the bounty where applicable.

I think both of you have valid points. There can be multiple roads to get to the same destination. Chill a bit guys, the more riled up you get the less work you're going to get done. People "shut down" when an argument gets heated, and I can already tell you neither of you are going to reach common ground if you keep goin at one another.

zk34911 - March 19, 2017, 10:52 am

(Mar 19, 2017 10:46 AM)Shoe Wrote: [\[CITE\]](#) I'd recommend taking some time to tidy up your blog and publish some content other than a brief portfolio. From a brief look at the page, I didn't believe you had any credentials what-so-ever to back up your bounty claims, but Twitter supplemented that. I'd recommend hyperlinking to the bounty where applicable.

There isn't any content on there as of now, and I don't plan to put any soon. It's just merely a portfolio of achievements. I am going to link my profiles soon

"From a brief look at the page, I didn't believe you had any credentials what-so-ever to back up your bounty claims"

>> I can't PM you right now, so that I could link you to my Bug Bounty profiles. My 3 last bug bounties have been \$1000+ from Twitter, you can check it on my Twitter/Instagram profiles which is down below on the blog. You will find your proof there.

"I'd recommend hyperlinking to the bounty where applicable."

>> You can check my Twitter/Instagram and I plan to edit my blog soon. Thanks

EDIT : Linked everything with my blog

Shoe - March 19, 2017, 10:57 am

(Mar 19, 2017 10:52 AM)zk34911 Wrote: [\[CITE\]](#) There isn't any content on there as of now, and I don't plan to put any soon. It's just merely a portfolio of achievements.

"From a brief look at the page, I didn't believe you had any credentials what-so-ever to back up your bounty claims"

>> I can't PM you right now, so that I could have linked you to my Bug Bounty profiles. Also my Twitter/Instagram username is down below on the blog. You will find your proof there.

"I'd recommend hyperlinking to the bounty where applicable."

>> You can check my Twitter/Instagram. Thanks

I meant on the blog. I was able to find proof on my own, however if Joe stumbles upon your blog, they may not be so inclined to check your accounts. Just a suggestion.

zk34911 - March 19, 2017, 11:06 am

(Mar 19, 2017 10:57 AM)Shoe Wrote: [\[CITE\]](#) I meant on the blog. I was able to find proof on my own, however if Joe stumbles upon your blog, they may not be so inclined to check your accounts. Just a suggestion.

Yeah, I think I need to edit it a little bit just to link my Bug Bounty profiles.

To anyone out there, if you want proof. You can PM me

EDIT : Linked everything with my blog

DoubleClickBTC - March 19, 2017, 11:14 am

Holy shit you two really went all out keyboardwarrior mode.

OP i recently did this course on [Udemy \(warez disallowed\)](#) There are many other great courses there aswell, im working through 4 at the moment, although not all are interactive but some do have a active QnA subforum. If you want to get the basics down in a week or less take a look at

[www.Udemy \(warez disallowed\).com/learn-website-hacking-penetration-testing-from-scratch/learn/v4/overview](#)

Although he does say no prior programming knowledge is required it will be in your best interest to go over to [www.codecademy.com](#) and do the PHP, Python and SQL courses. If you constantly dedicate your time you will be a mediocor pentester within a month or two. From there you will have a good base to take things to the next level

Nomad.Null - March 19, 2017, 8:10 pm

... Well then... I can see that this is not a friendly thread, lol.

@OP: I'll just throw in my recommendations for learning web exploitation. Personally, it's not my area of expertise, but I do happen to know plenty of people that are extraordinarily good at it and a handful of tools that they use to make it work for them. Before I start any of this, I want to point out that there is essentially only one reason why web vulnerabilities exist. Input validation and sanitation. Websites SHOULD be built on the premise not to trust the input that users give. The idea of exploiting websites is to find some place where the website allows you to inject your own code

into the server and get it to execute in one form or another. This could be Javascript, PHP, Python, Perl, C code, AJAX, doesn't really matter. If the server understands it and executes it, it is exploitable. Just bear this in mind as I explain everything going further...

For starters, if you want to learn web exploitation, you should start by getting familiar with website structure and how they are built. So, yes, HTML and CSS will at least help you understand how websites work as far as their structure and appearance. There are certain rare cases where the HTML allows you to actually execute code (ie: html injection), but this is not really exploitable. The main reason for learning HTML and CSS is to understand how websites are built and behave. It won't help you exploit it directly, but it is very important to understand. Glad we got that one out of the way...

Something also to keep in mind is that websites are very dependent upon directory structures for security purposes. Most web servers (apache and nginx for example) use something called an htaccess file. These are very important files, because they define the directory security of the website, as well as how it is rendered. If you fuck up the htaccess file (as an administrator) you can cause your whole website to go down. Because websites are organized in directory structures, htaccess is how the web server interprets which directories users are allowed to look at. It has a LOT of other features that I won't get into. You cannot exploit the htaccess file directly, but sometimes you can view it and this may sometimes reveal some very telling information, and may even tell you exactly where to start looking to begin exploiting the site. Other vulnerabilities like Directory Traversal allow you to take peeks around at the directory structure of the server.

Now, something else to keep in mind are the ways websites can be built for interactivity. By far, Javascript and PHP prevail as the two most popular languages for creating interactivity and logic with responsive websites. Personally, I feel like PHP is hot garbage as a programming language and is ugly as sin to look at, but nevertheless, it's quite popular. Javascript is a scripting language built into the web stack and supported by essentially every web server in order to add functions such as clickable buttons, conditional statements, and your basic logic that you get with any scripting language. It has some nice libraries and borrows its structure and style from C++. It is object-oriented and quite versatile, and therefore worth knowing.

Javascript is useful for ONE major reason in web exploitation. Client-side attacks. Client-side attacks, as the name suggests, are attacks which target the user browsing the website and involves using techniques which exploit vulnerabilities in the website's logic in order to get code to execute on the end-user's browser. Popular techniques include Cross Site Scripting (XSS), Cross Site Request Forgery (CSRF) and Click-Jacking. Each of these are a bit involved when you get into the details, but it should be mentioned that you do need to know Javascript in order to exploit them. The main reason why Javascript is only useful for client-side attacks is because it is a common language that is known to be compatible with all OS platforms and is useful for rendering dynamic, interactive content, so whenever it is easy to exploit a website with a client-side vulnerability, it is usually considered a very serious vulnerability.

PHP is used for an entirely different purpose and is never seen (directly) by the client. It works in the server backend in order to serve up data from elsewhere, such as a SQL database, in order to deliver it dynamically to the user's browser. PHP is capable of using much more complex data-types and is reserved entirely for server-side data processing. Typically, if you can inject PHP code into a website, you can interact with the operating system of the web server, meaning, you can get a shell, which grants you direct access to the server.

Finally there is SQL injection which is among the most serious as well as easiest vulnerabilities to exploit, but also one of the most noticeable from a defensive standpoint, especially if you use automated tools. Guys who are very skilled at SQL injection and know what to look for when

exploiting SQLi vulnerable sites can perform these kinds of attacks almost entirely with manual testing, but I digress. I mentioned that PHP code is used to interact with different data-types in the server backend. If certain input fields are not sanitized properly, you can inject SQL queries into the webpage which will cause it to execute SQL commands. SQL typically runs as the root, which leaves the most room for escalation. With SQL commands, you can dump information from the database, execute commands on the server, and even install a shell to gain yourself access to the system remotely. Keep in mind I am GROSSLY oversimplifying the process, but again, I don't have a lot of space here.

There are MANY more types of vulnerabilities to exploit, more than I can possibly expound upon in one post. What I highly recommend starting with is the Damn Vulnerable Web App at www.dvwa.co.uk, and go try the cybrary.it web exploitation course that was suggested above. It is a good start. The DVWA can be installed on a virtual machine with minimal resources, and gives you plenty of opportunity to practice exploiting realistic scenarios for vulnerable web applications and even provides tutorials and explanations for the exploits.

Tools I highly recommend for testing web applications are BurpSuite, Metasploit, Nikto, OWASP ZAP, Vega Web Vulnerability Scanner, SQLMap, and dirbuster. If you happen to have a Kali installation, all of these tools come installed. Cyborg and Black Arch Linux also come with these installed if you prefer. Otherwise, you'll need to install the tools manually on another Linux or, (if you dare) Windows system.

P.S. I do highly recommend Python as well. Can't say I am as militant about it as ByteKnight is, but I can definitely recommend it as a MUST HAVE for security professionals. Python is an extraordinarily powerful language, easy to read and write, powerful libraries, you can't go wrong. There are tons of features built into it, and even more tutorials available online for it. However, I highly recommend the Python for Security Professionals course on Cybrary.it

If you are going to learn Python, I suggest learning the latest version (Python 3.6). Python 2.7 and below are deprecated and no longer receiving updates to the libraries, and as it gets older, it will get much harder to find support, and you will find yourself always having to re-invent the wheel as the developers move all of their libraries over to Python 3.

Good luck!

Th4nat0s - March 20, 2017, 8:29 am

(Mar 19, 2017 08:10 PM)True Demon Wrote: [\[CITE\]](#) ... Well then... I can see that this is not a friendly thread, lol.

@OP: I'll just throw in my recommendations for learning web exploitation. Personally, it's not my area of expertise, but I do happen to know plenty of people that are extraordinarily good at it and a handful of tools that they use to make it work for them. Before I start any of this, I want to point out that there is essentially only one reason why web vulnerabilities exist. Input validation and sanitation. Websites SHOULD be built on the premise not to trust the input that users give. The idea of exploiting websites is to find some place where the website allows you to inject your own code into the server and get it to execute in one form or another. This could be Javascript, PHP, Python, Perl, C code, AJAX, doesn't really matter. If the server understands it and executes it, it is exploitable. Just bear this in mind as I explain everything going further...

For starters, if you want to learn web exploitation, you should start by getting familiar with website structure and how they are built. So, yes, HTML and CSS will at least help you understand how websites work as far as their structure and appearance. There are certain rare cases where the HTML allows you to actually execute code (ie: html injection), but this is not really exploitable. The

main reason for learning HTML and CSS is to understand how websites are built and behave. It won't help you exploit it directly, but it is very important to understand. Glad we got that one out of the way...

Something also to keep in mind is that websites are very dependent upon directory structures for security purposes. Most web servers (apache and nginx for example) use something called an htaccess file. These are very important files, because they define the directory security of the website, as well as how it is rendered. If you fuck up the htaccess file (as an administrator) you can cause your whole website to go down. Because websites are organized in directory structures, htaccess is how the web server interprets which directories users are allowed to look at. It has a LOT of other features that I won't get into. You cannot exploit the htaccess file directly, but sometimes you can view it and this may sometimes reveal some very telling information, and may even tell you exactly where to start looking to begin exploiting the site. Other vulnerabilities like Directory Traversal allow you to take peeks around at the directory structure of the server.

Now, something else to keep in mind are the ways websites can be built for interactivity. By far, Javascript and PHP prevail as the two most popular languages for creating interactivity and logic with responsive websites. Personally, I feel like PHP is hot garbage as a programming language and is ugly as sin to look at, but nevertheless, it's quite popular. Javascript is a scripting language built into the web stack and supported by essentially every web server in order to add functions such as clickable buttons, conditional statements, and your basic logic that you get with any scripting language. It has some nice libraries and borrows its structure and style from C++. It is object-oriented and quite versatile, and therefore worth knowing.

Javascript is useful for ONE major reason in web exploitation. Client-side attacks. Client-side attacks, as the name suggests, are attacks which target the user browsing the website and involves using techniques which exploit vulnerabilities in the website's logic in order to get code to execute on the end-user's browser. Popular techniques include Cross Site Scripting (XSS), Cross Site Request Forgery (CSRF) and Click-Jacking. Each of these are a bit involved when you get into the details, but it should be mentioned that you do need to know Javascript in order to exploit them. The main reason why Javascript is only useful for client-side attacks is because it is a common language that is known to be compatible with all OS platforms and is useful for rendering dynamic, interactive content, so whenever it is easy to exploit a website with a client-side vulnerability, it is usually considered a very serious vulnerability.

PHP is used for an entirely different purpose and is never seen (directly) by the client. It works in the server backend in order to serve up data from elsewhere, such as a SQL database, in order to deliver it dynamically to the user's browser. PHP is capable of using much more complex data-types and is reserved entirely for server-side data processing. Typically, if you can inject PHP code into a website, you can interact with the operating system of the web server, meaning, you can get a shell, which grants you direct access to the server.

Finally there is SQL injection which is among the most serious as well as easiest vulnerabilities to exploit, but also one of the most noticeable from a defensive standpoint, especially if you use automated tools. Guys who are very skilled at SQL injection and know what to look for when exploiting SQLi vulnerable sites can perform these kinds of attacks almost entirely with manual testing, but I digress. I mentioned that PHP code is used to interact with different data-types in the server backend. If certain input fields are not sanitized properly, you can inject SQL queries into the webpage which will cause it to execute SQL commands. SQL typically runs as the root, which leaves the most room for escalation. With SQL commands, you can dump information from the database, execute commands on the server, and even install a shell to gain yourself access to the system remotely. Keep in mind I am GROSSLY oversimplifying the process, but again, I don't have a lot of space here.

There are MANY more types of vulnerabilities to exploit, more than I can possibly expound upon in one post. What I highly recommend starting with is the Damn Vulnerable Web App at www.dvwa.co.uk, and go try the cybrary.it web exploitation course that was suggested above. It is a good start. The DVWA can be installed on a virtual machine with minimal resources, and gives you plenty of opportunity to practice exploiting realistic scenarios for vulnerable web applications and even provides tutorials and explanations for the exploits.

Tools I highly recommend for testing web applications are BurpSuite, Metasploit, Nikto, OWASP ZAP, Vega Web Vulnerability Scanner, SQLMap, and dirbuster. If you happen to have a Kali installation, all of these tools come installed. Cyborg and Black Arch Linux also come with these installed if you prefer. Otherwise, you'll need to install the tools manually on another Linux or, (if you dare) Windows system.

P.S. I do highly recommend Python as well. Can't say I am as militant about it as ByteKnight is, but I can definitely recommend it as a MUST HAVE for security professionals. Python is an extraordinarily powerful language, easy to read and write, powerful libraries, you can't go wrong. There are tons of features built into it, and even more tutorials available online for it. However, I highly recommend the Python for Security Professionals course on Cybrary.it

If you are going to learn Python, I suggest learning the latest version (Python 3.6). Python 2.7 and below are deprecated and no longer receiving updates to the libraries, and as it gets older, it will get much harder to find support, and you will find yourself always having to re-invent the wheel as the developers move all of their libraries over to Python 3.

Good luck!

Thanks man, I really appreciate it :) ?