

Forum:[Hack Forums](#)

Thread: [How do people actually learn programming/hacking?](#)

Ihatemyphone - November 30, 2017, 1:06 pm

Did you guys start with tutorials? A friend taught you? Just messing around trying different shit? I'm genuinely curious but having trouble finding a middle ground between a boring tutorial (2+2=4) and an extremely technical tutorial that is way over my head. People with experience, how did you learn?

CyberStrike - November 30, 2017, 1:14 pm

For me I read books and did my own labs, to practice what I have read and learned.

SmoX - November 30, 2017, 1:15 pm

Read some books, online pdf's and questions you have, google them, thats how you learn, the problem everyone thinks.. is if you learn how to hack you then know everything.. No, when people learn hacking/programming etc they don't stop learning, its ever changing

Faith Connors - November 30, 2017, 2:00 pm

read
understand what you read
apply what you read to different topic and try to figure out why it matters
practice what you learned in an actual lab.

NoWay4h - November 30, 2017, 2:09 pm

...Sitting thousand of hours before their computer, google, try & practice what they want to learn. You have to know the direction you want to go, but first it's important to know that there's much more than just programming^^ and that this is a life if you want to get good at it and that it isn't learned from today to tomorrow. Go your way, get a solid fundament on knowledge and then search your path. I'm recommending A+ & Network+ before everything else.

JillOk - November 30, 2017, 2:13 pm

Read, Watch Tutorials, and try making own small programs with what you know.

splutter - November 30, 2017, 2:22 pm

It's the same as learning anything else. Theoretical fundamentals -> practical application -> understanding.

Pick up a book or many books which you may find by using Google. Start understanding the basics and the necessary theory - this is probably the most boring part so you better get some kind of motivation. For hacking, coding should be part of your basics so you can develop an appreciation at the implementation level and also because it allows you to code your own tools if there aren't any already existing ones for your needs. If the learning material is too hard, find out why, put it on hold, then go research more before coming back to it. When you've built your fundamentals, apply it in practice and most importantly, experiment.

Armada - November 30, 2017, 4:30 pm

(Nov 30, 2017 02:06 PM)Ihatemyphone Wrote: [\[CITE\]](#) Did you guys start with tutorials? A friend taught you? Just messing around trying different shit? I'm genuinely curious but having trouble finding a middle ground between a boring tutorial (2+2=4) and an extremely technical tutorial that is way over my head. People with experience, how did you learn?

Start picking them technical books up.

I got 2 shelves worth of technical books, and I am CONSTANTLY swapping the outdated ones out with new ones.

Put aside the white papers, and other technical write ups, It is a daily learning lesson.

AnomalyX - November 30, 2017, 6:14 pm

Reading has been extremely helpful for me. I am in no way an expert, but books helped me to break things down and better understand how things function at a basic level

Hydr - December 1, 2017, 8:58 am

Well.. reading is the only option bro, i guess you need some achievements and goals to start learning what you need ;)

Dxnboi - December 1, 2017, 5:45 pm

I recommend you just learn how to code and debug first before getting too far. Thats where I started. Try learning Java and C++

eth1 - December 14, 2017, 10:30 pm

I watched many YouTube tutorials, went on forums, asked friends, but most importantly - just practicing.

Start by learning doxing or phishing for example.

Then move on to RATs and malware.. Then Exploits.. So on

dumbrascal6969lol - December 14, 2017, 11:02 pm

A good place to start is installing a distro like Kali, going over the tools list, figuring out what they all do, and just read the man/-h pages for them. Then grab an easy lab or two, and try things out. If they you still can't get it, there's hundreds of tutorials on youtube that'll help.

Nomad.Null - December 15, 2017, 7:09 am

(Dec 15, 2017 12:02 AM)dumbrascal6969lol Wrote: [\[CITE\]](#) A good place to start is installing a distro like Kali, going over the tools list, figuring out what they all do, and just read the man/-h pages for them. Then grab an easy lab or two, and try things out. If they you still can't get it, there's hundreds of tutorials on youtube that'll help.

I disagree with diving head-first into what is possibly the least beginner-friendly Linux distro on the planet as a way of learning how to hack. 90% of the tools in the Kali distro are unused or have very specific and/or highly technical use-cases.

You SHOULD download and install Kali as a virtual machine so that you have those tools available to you, but what you need to do is start reading, like everyone has been saying. There are a few books I recommend that you can start with, and begin working in some labs/ctf environments. The learning process for hacking largely consists of repetition and persistence, literally beating your head against a brick wall until it finally sticks.

These are my recommendations:If you want to start learning Linux for learning Linux's sake, start with Ubuntu, Mint, or Fedora. There are many flavors out there, but these are the most user-friendly, beginner-friendly, and best supported out there. Kali is based off the Debian distro, which is one of the oldest and most stable, but is very minimalistic and better-suited for building servers. It also does not have extensive driver support, and I frequently ran into problems with obscure driver issues while using it . Kali comes with way too many tools and you will quickly become overwhelmed. You can download it as a VM, but don't just go through the list of tools one-by-one to learn them. That would be tedious, entirely unhelpful to you, take way too long, and is just insane.

Check out these books: The Hacker Ethos (I wrote this for beginners), [The Hacker Playbook 2](#) by Peter Kim, [Metasploit Unleashed](#) (Free), [Penetration Testing: A Hands-On Introduction to Hacking](#) by Georgia Weidman

Video Tutorials for penetration testing can be found on <https://www.cybrary.it/>

Use Google to research about vulnerabilities and advanced concepts.

Reference exploit-db.com for reliable, safe exploit code that has been reviewed and approved by the Offensive Security team. It will take you a while before you understand or are ready to use any of this stuff, but I want you to keep it in the back of your mind because it's a good resource.

Learn a programming language: I recommend starting with something beginner-friendly with good documentation and plenty of powerful code libraries to use. For this, I suggest Python (3). You will also benefit from learning the Linux BASH scripting language, which allows you to create tools that interact with the Linux environment. Here are some books/resources to try: Blackhat Python, Python for Penetration Testers, [Bash Beginner's Guide](#), readthedocs.org (Amazing code documentation for EVERYTHING)

Build a hacking lab of your own using machines from vulnhub.com

Join some public Capture the Flag events/sites. Some I recommend are <https://www.hackthebox.eu>, <https://lab.pentestit.ru>, and <https://hackthissite.org>

"Advanced Hacking & Programming tips" ([Click to View](#))

I want to warn you ahead of time about the difference between interpreted (scripting) languages, and compiled languages. Python is an interpreted language...very high-level, easy to learn, easy to use, and suitable for writing tools and automating tasks, but they usually not as fast as compiled languages and cannot or are not meant to interact with the hardware of a computer. Compiled languages (such as C) tend to be more strict, lower-level (Low level means it works directly with or very close to the hardware level), requires a lot more code to do simpler things, and are much more difficult to learn; however, they have the benefit of being much faster than interpreted languages, allow you to interact directly with the hardware, and are much more difficult for someone to disassemble, meaning they cannot easily read and understand what your code is doing or how it was built.

Compiled languages are usually what malware is built with. Additionally, understanding how to read and write code in a compiled language, and also understanding the Assembler language (machine code) will enable you to understand how programs work and interact with the hardware. This is very powerful for a hacker because someone who understands what a program does at the machine level enables them to see flaws in the programs themselves. This is what leads to discovering bugs and building exploits. If you recall the WannaCry Ransomware incident in May, that was because of a bug in the Microsoft Windows Operating System that has existed since 2000. The NSA found it at some point, but nobody else did until 2017 when it was broadcast to the world by the Shadow Brokers. If you want to get to that level, then once you are ready, read Hacking: The Art of Exploitation and The Rootkit Arsenal: Escape and Evasion in the Dark Corners of the System.

You need to be willing to dive into the highly technical material and very boring reads. The fundamentals of hacking are the most important concepts to learn. You can anticipate being bored with a lot of things you read, but occasionally you encounter some gems. You will unfortunately have to go tapping into the Google mine of bullshit to eventually pull out those diamonds.

For some additional help, here are some blogs and RSS feeds I recommend subscribing to and reading through for some pretty awesome tips on hacking and other news.

FuzzySecurity.com

Hackaday.com

<http://blog.gentilkiwi.com/> (In French, use Google to translate. It's worth the trouble, trust me.)

<https://warroom.securestate.com/>

<https://wired.com>

<https://www.darkreading.com>

<http://ethicalhackingblog.com/>

That should be plenty of material to keep you busy for years. Good luck. ?

Armada - December 15, 2017, 7:17 am

(Dec 15, 2017 08:09 AM)True Demon Wrote: [\[CITE\]](#) I disagree with diving head-first into what is possibly the least beginner-friendly Linux distro on the planet as a way of learning how to hack. 90% of the tools in the Kali distro are unused or have very specific and/or highly technical use-cases.

You SHOULD download and install Kali as a virtual machine so that you have those tools available to you, but what you need to do is start reading, like everyone has been saying. There are a few books I recommend that you can start with, and begin working in some labs/ctf environments. The learning process for hacking largely consists of repetition and persistence, literally beating your head against a brick wall until it finally sticks.

These are my recommendations: If you want to start learning Linux for learning Linux's sake, start with Ubuntu, Mint, or Fedora. There are many flavors out there, but these are the most user-friendly, beginner-friendly, and best supported out there. Kali is based off the Debian distro, which is one of the oldest and most stable, but is very minimalistic and better-suited for building servers. It also does not have extensive driver support, and I frequently ran into problems with obscure driver issues while using it. Kali comes with way too many tools and you will quickly become overwhelmed. You can download it as a VM, but don't just go through the list of tools one-by-one to learn them. That would be tedious, entirely unhelpful to you, take way too long, and is just insane.

Check out these books: The Hacker Ethos (I wrote this for beginners), [The Hacker Playbook 2](#) by Peter Kim, [Metasploit Unleashed](#) (Free), [Penetration Testing: A Hands-On Introduction to Hacking](#) by Georgia Weidman

Video Tutorials for penetration testing can be found on <https://www.cybrary.it/>

Use Google to research about vulnerabilities and advanced concepts.

Reference exploit-db.com for reliable, safe exploit code that has been reviewed and approved by the Offensive Security team. It will take you a while before you understand or are ready to use any of this stuff, but I want you to keep it in the back of your mind because it's a good resource.

Learn a programming language: I recommend starting with something beginner-friendly with good documentation and plenty of powerful code libraries to use. For this, I suggest Python (3). You will

also benefit from learning the Linux BASH scripting language, which allows you to create tools that interact with the Linux environment. Here are some books/resources to try: Blackhat Python, Python for Penetration Testers, [Bash Beginner's Guide](#), [readthedocs.org](#) (Amazing code documentation for EVERYTHING)

Build a hacking lab of your own using machines from [vulnhub.com](#)

Join some public Capture the Flag events/sites. Some I recommend are <https://www.hackthebox.eu>, <https://lab.pentestit.ru>, and <https://hackthissite.org>

"Advanced Hacking & Programming tips" ([Click to View](#))

I want to warn you ahead of time about the difference between interpreted (scripting) languages, and compiled languages. Python is an interpreted language...very high-level, easy to learn, easy to use, and suitable for writing tools and automating tasks, but they usually not as fast as compiled languages and cannot or are not meant to interact with the hardware of a computer. Compiled languages (such as C) tend to be more strict, lower-level (Low level means it works directly with or very close to the hardware level), requires a lot more code to do simpler things, and are much more difficult to learn; however, they have the benefit of being much faster than interpreted languages, allow you to interact directly with the hardware, and are much more difficult for someone to disassemble, meaning they cannot easily read and understand what your code is doing or how it was built.

Compiled languages are usually what malware is built with. Additionally, understanding how to read and write code in a compiled language, and also understanding the Assembler language (machine code) will enable you to understand how programs work and interact with the hardware. This is very powerful for a hacker because someone who understands what a program does at the machine level enables them to see flaws in the programs themselves. This is what leads to discovering bugs and building exploits. If you recall the WannaCry Ransomware incident in May, that was because of a bug in the Microsoft Windows Operating System that has existed since 2000. The NSA found it at some point, but nobody else did until 2017 when it was broadcast to the world by the Shadow Brokers. If you want to get to that level, then once you are ready, read Hacking: The Art of Exploitation and The Rootkit Arsenal: Escape and Evasion in the Dark Corners of the System.

You need to be willing to dive into the highly technical material and very boring reads. The fundamentals of hacking are the most important concepts to learn. You can anticipate being bored with a lot of things you read, but occasionally you encounter some gems. You will unfortunately have to go tapping into the Google mine of bullshit to eventually pull out those diamonds.

For some additional help, here are some blogs and RSS feeds I recommend subscribing to and reading through for some pretty awesome tips on hacking and other news.

[FuzzySecurity.com](#)

[Hackaday.com](#)

<http://blog.gentilkiwi.com/> (In French, use Google to translate. It's worth the trouble, trust me.)

<https://warroom.securestate.com/>

<https://wired.com>

<https://www.darkreading.com>

<http://ethicalhackingblog.com/>

That should be plenty of material to keep you busy for years. Good luck. ?
Just list the proper Book already ?
<https://hackforums.net/showthread.php?tid=5102103>

TrueDemon is one of the FEW people you can actually take advice from without worry its gonna be fucked up.

I STRONGLY recommend his book for anyone who is starting in the field, he had me proof read it, and it is fucking top notch.

@OP take what True has to say to heart, he will not mislead you in anyway. ?

[BLANK] - December 15, 2017, 9:36 am

Start with the foundational information:

Networking
Web Applications
Basic Programming fundamentals

Then find an area that really interests you and specialize in it. If you have money I recommend "elearnsecurity" site, they have a lot of great courses of different levels. I was able to prepare for the OSCP using their courses, they are more hands on then PWK. You also have a site like "pentesterlab" which is great for learning web application hacking. Cybrary.it is also a great resource. I also recommend "hackthebox.eu" because its free and is very similar to the PWK labs you pay for when preparing for OSCP.

So you don't have to just read read read, if you prefer videos there are many good online courses, some free. In the end though it comes down to persistence. You have to set up labs (VMs) and practice as much as you can. You don't have to be an expert programmer but you should understand coding and scripting and basic computer science principles. As you advance you will pick up more programming anyways, but the most important thing is that you could adapt a script to a situation and modify as needed and not just get stuck if the default script doesn't work.

So the point is, pick a topic that really interests you and focus on it. Being too general is difficult and will take you a lot longer to get good at any one thing.

American Ninja - December 15, 2017, 10:09 am

start reading books . Ebooks and research

start watching tutorials

learn to hack by practicing it over and over again even when you fail never give up

Mr.toboR - December 15, 2017, 12:43 pm

I was exactly like you and as they said start with good books and I recommend starting with comptia A+ book that you can buy or find it for free with a little search in google ? . And I recommend also the great real hacking books series : Hacking exposed series , yeah those are my own favorites so far and you can find them for free with little seach also. And i also recommend trying learning how to code and you can try c++ or java but for real try to code with python because it's good for hacking ,easy to learn and good luck ?.

SIM-3063187 - December 15, 2017, 12:46 pm

You can watch 'thenewboston' on YouTube. Just start by learning Python

AU Subi - December 15, 2017, 1:01 pm

(Nov 30, 2017 02:06 PM)Ihatemyphone Wrote: [\[CITE\]](#) Did you guys start with tutorials? A friend taught you? Just messing around trying different shit? I'm genuinely curious but having trouble finding a middle ground between a boring tutorial ($2+2=4$) and an extremely technical tutorial that is way over my head. People with experience, how did you learn?

Online courses, books, or both, depending on your learning style. Don't look for programming tutorials on YouTube, find a site with an actual start to finish course, e.g learncpp.com. Hate to break it to you, but you're going to have to learn these extremely fundamental and concrete operations like $2 + 2 = 4$ first. You will need to do really boring projects before you have built enough knowledge to create something interesting. However, if you are programming in C/C++, definitely check out the Windows API once you have a solid understanding of the basics. It contains functions that allow you to perform more significant operations like copy to clipboard, check if key is pressed, etc, in very little code.

Xerotic - December 15, 2017, 1:10 pm

I started by watching Python tutorials and slowly just kept trying to make cooler and cooler scripts. Then I learned about web technologies because I wanted to mess with MyBB plugins. By altering and hacking a bunch of plugins to do what I want, I began to understand more and more.

So basically a combination of tutorials and then trying to alter other code already made to learn better.

CyberStrike - December 15, 2017, 7:33 pm

Look if you want at some point to use linux, start out easy,even using a VM is an option to learn about linux. Ubuntu is simple enough I have been using linux for a while and I at the time started out with Slackware 7.0 circa 1999, back then being brand new it was far from the easiest thing, it wasn't like how it is now we would have to pretty much compile everything go through dependency hell then finally it would either compile or it would error and we got more work to do. Also if you needed help you got on IRC, where if you had any sort of GUI guess what you were called and considered as a skid then searching on the internet may not produce a result. You had to wade through manuals written by and for developers. Oh yeah and kernel upgrades, you had to have a priest by your side and during that initial reboot you would make sure that the Pope put in a good word for you so the kernel panic demons stay away. While all of this is going on I also had to make sure I have my assignment ready every Friday.

Jumping straight into Kali and using a swim or sink environment is easy to dissuade a new user.