While participating in a security conference, I was asked a very important question: "How do we protect ourselves from Zero Days?". My answer: "There is no such thing as zero days". Crazy right? How can that be true? People report "zero days" everyday, so how can that statement be true? And how does that answer solve the problem? Fear not fair reader, all will be revealed! Lets dig a little deeper and find why there may not really be *"zero day"* vulnerabilit ies and why protecting against them isn't as hard you may think.

A trip down memory lane

First a little history about the term "zero day". A long time ago, when dinosaurs ruled the earth and men were men, as an old Army friend of mine likes to say, a great invention called the direct connect modem was created. For those of you too young to remember, please induge me for a moment as a wax poetic about this marvelous piece of technology that a skrimped and saved to buy as a young man.

The direct connect modem was a watershed moment for the PC world. It changed everything. Aside from being "better" than the old acoustic modem, it was also faster. It let us all, well, directly connect our computers to a phone line, and from there to other computers. Oh it was terribly by todays or even yesterdays standards, but to anyone that had one way back then it was marvelously fast! Why we could communicate at a blazing 300 baud! And aside from BBSing and logging into boxes at the local university so we could play on the proto-internet, we could even download the simple programs we manually typed, by hand, from computer magazines and share them with each other! What a time saver! It was Glorious!

Around that time software was also being sold for PCs, and well most of us young folks couldn't afford some of that software. So, along with this marvelous revolution came the so called "warez" community. People that published software that they didnt type in themselves and were maybe illegally or just unethically sharing these commercial programs with others. I won't judge, it was wild times.

Because modems were so slow the idea of a new piece of "purloined software" being published the day the publisher released it by the "warez" community seemed impossible, if you wanted something new you bought it, otherwise you got a copy from someone that had later and you checked your morals at the door. Eventually this started to happen before the publisher

published it. This was consider truly to be the work of a dark wizard. Commercial software started getting "released" by "warez" groups before the publisher could even get it in stores. Such creatures called these "zero day releases", meaning it was released not a week or a month after a piece of software was officially published, but either on the same day or even before. Magic!

Around the same time, people started to learn to really tinkerwith , or hack, computers. Eventually this included learning to break into computers and sharing this information with others. Sometimes for good and sometimes for bad reasons. The good guys initially reported vulnerabilities to their vendors, quietly, and the good vendors put out patches. Some vendors, however, started to take the position that if a vulnerability wasn't "known" to the public (i.e. their users) or wasn't "being actively exploited" then it was just "theoretical" and therefore didn't need to be fixed. From this, the "full disclosure" community was born. Security researchers felt the need to let others know about these vulnerabilities, both to warn people so they could take their own actions to defend against it and in some cases to pressure the vendors into fixing these issues. Eventually, the vendors started to claim these public announcements were "theoretical" too, so then research started putting out "proof of concept exploits" to prove that a vulnerability was real, and not just theoretical. Often a vulnerability would be published along with an exploit to really drive the point home. In parallel, the bad guys starts to write their own exploits, but they kept them a secret except to other bad guys.

Well, the "warez" and the "cracker" or unfortunately also known by then, and now, as "hacker" communities were a bit like an overlapping venn diagram even way back then. Not everyone in each community was in both communities, but some where and they definitely cross polinated their unique vernaculares. So the term "zero day" also started to be used by security researchers, exploit writers and everyone inbetween. This term meant a new vulnerability was so screamingly "new" that it was a "zero day" and a patch didnt exist. It many cases it was "released" before the vendor even knew about it, and generally this came with a fairly easy to use exploit for the vulnerability. So, these things were a big deal. So, the term stuck, and its even become so mainstream in the security and IT communities that I've heard it used in management and regulatory circles by non-cyber people.

OK Mike, thanks for the history leason, what does this have to do with zero days not existing and protecting my systems? You just proved that they do exist, can I have my 5 minutes back? And please don't tell me your entire argument is just semantics.

Well, first, thanks for your patience as I went down memory lane. You see I think its important to understand what "zero day" means before we can start to explain how the term is overhyped

and isnt the big deal some marketing folks make it out to be. From there, you can make informed decisions about how you can protect your systems wisely, and not waste your time and resources on solutions that are pure hype.

How Securiity is sold

Fear. Security is sold via fear. Vendors scare people, and then offer a solution to that fear. It works. A zero day, in common parlence, means: a new vulnerability that is published before a fix is available from the vendor. In short, it means there isnt a simple patch from the vendor to make the vulnerability go away. Scary stuff right?

No, its not. In fact in most cases its not even news. Blasphemy! I've been told by some of my peers I'm just dead wrong. Zero Days are real and they represent a terrible threat to life, liberty and the pursuit of happiness! You're wrong Mike, you're wrong!

Well, I understand the value that marketing has on what people think and I'm in the security industry too. They believe this because "Zero Day" has come to mean something that it actually isn't: The marketing people have convinced people that what "zero day" means is that someone published a brand spanking new ***method*** for compromising systems and we can't protect ourselves from it. Most, and I mean it, almost every single one of those so called "zero days" is not a new ***method*** and you ***can already defend against them***. Not so scary now eh? Most of these"zero day vulnerabilities" use the same methods that every other so called "zero day exploit" uses, so they aren't a new method.

OK Mike thanks for the english lesson, but I'm still vulnerable right? Maybe not.

The real threat

Lets break this down. There are really two types of "new" security vulnerabilities:

(1) those using "known" methods and

(2) those using "unknown" new methods.

The former is just that, a new flaw or vulnerability in some application, that is **using a method we already understand and can defend against**
(shell injection for example) and the later is using
a new method can not defend against
. Now the later could be a big deal, but its always a big deal. It may be that we just thought it wasnt feasible to exploit or use the method, but sometimes its the real deal and its something we never thought of at all or completely dismissed as "theoretical".

An example of a "new" vulnerability, or a method we already understand and can defend against, is a "sql injection attack". We know all about them, and we know how to detect and prevent them. We don't always do that, but the fact that we don't do something and that we don't know what to do are two totally different things. The failure to act does not mean a problem is unsolvable, it just means we didn't act especially if we know what to do!

An example of the later, a brand new method, would be when shell code injection was first "discovered" over twenty years ago (god I feel old saying that). Twenty years ago, shell code injection would be a "zero day method" because we didn't really consider them vulnerabilities, or we didn't think it was possible for a bad guy to perform them so we didn't do anything about them (nor did we really have a feasible solution to them short of "dont write bad code"). Because we just didn't think much about them, we didn't implement countermeasures to stop them, and because they could be performed that made every new bug that allowed a shell code injection into a true "zero day". Short of a patch from the vendor, you were in deep trouble.

Nowadays we have countermeasures for this. And some operating systems come with them, which helps to prevent this method from being performed. In essence, this protects us from the method, so even if an application has a bug like this the system may be immune to the method (and in good implementations it generally is). Keep in mind some OSes countermeasures are better than others in this case, some of them are unfortunately pretty weak. Nevertheless, if you have an effective implementation, then you have a good countermeasure to a method.

The Real Stuff

Now on to the real "zero day" stuff. That is: a brand new method of compromise. Truly zero day methods are *rare*, only a handful of them may be discovered over a few years. Please don't take this to mean that most cyber security efforts are therefore adequate to defend against the bad guys, the truth is most of them are woefully inept at protecting against even methods we already understand and should know better. But always remember, its about methods, not raw numbers of vulnerabilities. Some vendors misrepresent what their products can do by claiming large numbers of things they protect against to "stop the latest things", when in reality all there are stopping are really old things. Sometimes methods so old that any security vendor should be ashamed to say is "new".

Whats really interesting about "new methods " is many of them may also be prevented by countermeasures we use for other methods. Sometimes this is purely accidentally, but other times its by design. For example, lets say we knew all about SQL injection attacks, but had never thought of cross site scripting. One way we can protect systems from SQL injection attacks is with "input validation". This is a method, sometimes referred to as "positive security" or "whitelisting" where we define the known safe non-malicious inputs an application accepts and we reject anything else. Kind of like a firewall with an "unless allowed, deny" configuration. Lets say we defined all the known non malicious inputs into a web application, this can also protect the application against cross site scripting. And part of this is because some new "methods" use the same vectors to accomplish their goals as older methods.

Or you might build your enterprise around the fact that you will have vulnerable software (BTW, you will, dont let anyone fool you, you will have vulnerable software, and probably hardware too). You might also assume you have malicious users and as a result you build in really good monitoring capabilities. You define traffic flow patterns based on the known safe activity of your users, and anything that deviates from that sounds an alert. You might not stop the malicious software initially, but if you have a good response plan, response capability (as in 24/7 SOC and qualified people ready to respond 24/7) you may be able to manage the impact of a compromise such that its within your acceptance range.

The real real stuff

Which brings us to the really real "zero days", or a brand spanking new method that we have no idea how to defend against, or maybe we dont even know how to detect. These are rare,

but real. The key here is to ask "is this one of those times?". Most of the time, the marketing roar will try to convince you that it is, remember these are rare. Most of the time, when people claim something is new, its usually not. So first, ask questions, the odds pretty high that this isnt a new method, and you can already do something about it.

Now, what do you do if this is an honest to goodness real zero day method? When we have cases where we've got a truly new method that we don't have any response to, this is when your security program earns its pay. This is where defense in depth can save you, this is where listening to your security people in advanced will pay off in spades. We'll talk more in part II about what you can do about these, but remember, these are rare. These are the important things you should care about, but dont get caught up in the marketing hype. Most of the time when people say something is "new", what they mean is its using an old method and there isnt a patch for it yet.

I'll give you a little advanced peak . The key here is to both have a plan to adapt your technology investments and a plan to recover if you do get compromised.

More on this in Part II of this blog.

Why its not as crazy as it seems

A lot, and I mean a lot of the vulnerabilities that are published and called "Zero Day" are using methods to compromise systems that we already understand. And because we already understand the ***method*** that means we can effectively defend effectively against them. And heres the important part: ***even though we may not know about the specific vulnerabilities in our systems, if they are using known methods we can defend against them!***

Thats right. Even though you have vulnerable applications, operating systems and so - if you build your security programs and counter measures around known methods and known root causes of compromise, you can stop what most people call "zero days vulnerabilities".

The point of this post is that "zero day vulnerabilities" are not the unsolvable menace the hype has made them out to be, they aren't the problem, its the methods. Most of the time, you can already defend against the methods, so most of these zero days are non-events (if you are defending against the methods).

And in those rare cases where you don't have a way to counter the method, if you implement a good security program (I promise we will talk about how to do that too in part II!), thats designed to address known and systemic root causes of compromise you can prevent the methods those "zero day vulnerabilities" use from doing unthinkable damage.

Bad things will happen, plan for it, but dont panic its not as bad as you think! Think of the spare tire in your car, you have one, its OK to drive your car! So, if you are doing the right things you won't need to panic the next time someone tells you about yet another "zero day" using a method you already protect against. Instead, you can focus your security efforts on new methods, and enhancing your program to provide adqueate protection against the threats your assets will face.

So back to my statement: "There is no such thing as zero day vulnerabilities". As you can see, its not the "vulnerabilities" that matter, its the methods. Its the how, not the what. If we can defend against how compromises occur, the root causes, we can make "zero day vulnerabilities" a non issue.

In part II of this article we'll get into the details of how you defend against zero day methods.