CyberFestCarousel.docx

14/02/2017

Introduction

This document outlines some scripts that can be used to deliver outreach content to people between the ages of 13 and 17. The content focuses on basic "cyber" related information. "Cyber" in this context is defined as a broad term encompassing such things as how the Internet works, how to stay safe online and some basic cyber-attacks.

The format for delivery is intended to be a carousel of workshops. However, these workshops are all capable of being run independently. For younger age groups there is another package of activities that are very similar to these (CyberFest Activity Pack), but they are intended to be delivered in a story like format and are not quite as technical.

Checklist

This checklist should be used to ensure that you have gathered all of the equipment that is necessary to complete the workshops. This list covers just one instance of each workshop. For larger audiences, you could run several instances of each workshop and for this you will require more equipment.

Item	Check
Passwords	
Passwords and phrases laminated cards	
Whiteboard	
Selection of board pens	
DDOS	
Bag of coloured plastic balls	
3 buckets	
Packetisation	
Bag of coloured plastic balls	
Large board with webpage on it	
Bucket	
TCP/UDP	
Large picture cut into 10 or so pieces	
Large picture cut into 10 or so pieces and numbered on the back in sequential order.	

Passwords

Introduction

The purpose of this workshop is to get across the concept that a strong password is required and that remembering a strong password needn't be that difficult. The concept presented here is to use a well-known phrase or song lyrics or similar and take just the first letter from each word. The convert some of these characters to upper or lower case, or to a representative special character or number (e.g. an 's' could be a '$').

Script

Activity	Script	Timings
Intro	What is the most important thing to you in your home? Would you agree that it wouldn't be a good thing if someone broke into your house and took it, damaged it or replaced it with an imitation? What is currently stopping someone from doing that (wait for responses like 'locks')? You might have different strengths of locks depending on what you are trying to protect. For example, you might put a simple Yale lock on your front door because it mixes reasonable strength with convenience. Whereas you might put a vault door requiring three people to open it on a room storing gold bullion. When it comes to your accounts online – like Facebook and your banking – what gives your lock strength (*we are looking for 'passwords'*)? So just like with your house containing your important things, you are going to want a lock that is effective. Something that cannot easily be picked. In this workshop, we will take a quick peak at strong and weak passwords and give you a tip about how to easily remember complex passwords.	2 mins
Password crack	To show you how easy it can be to crack a bad password, we will have a go at it right now. I have a password in mind and you are all going to perform a brute force attack on it. A brute force attack is not particularly sophisticated. It just means you are going to try lots of different words until you get it. To start you off, this is the hint you get when you click on the 'need a hint' button at the login screen: fruit+year. It might be easiest to start with the year. What might be an important year to me (*try to steer them away from actual digits and think about the occasion and why I might remember it. In this case, use your birth year*)? Now that you have a handful of dates that you could choose from,	5 mins

	you have a quick look at my Facebook page and see that my birth year is 1990. It's a fair assumption that this is probably the year used in this bad password. Next let's try to crack the fruit bit. What could we try (*simplest answer is to just try different fruits with the year at the end*)? Give me some fruits then and if you get the right one you're in. (*Just pick a fruit at random that you will use for this. In the interest of time, you could just take the 3rd or 4th answer that is shouted out and pick that one*). Congratulations, you successfully cracked my password without too much effort in approximately 2 minutes. You didn't need any special programs to do it either	
Analysis	So what were the good things about this password (*we are looking for 'at least 8 characters long' and 'mix of numbers and letters'*)? What wasn't so good (*we are looking for 'no special characters', 'no uppercase', 'real word used' and 'personal information included'*)? So just from this very quick analysis we can see that a strong password should contain at least 8 characters – the fastest super computer in the world would take 20,000 years to cycle through every combination of 8 characters – and those characters should be a mix of upper and lower case, normal and special characters and numbers, and should not contain any actual words or personal information (*such as year of birth or name pf parent etc.*).	2 mins
Pro tip	So as you can see, with all of these rules, your password will be very complex and difficult to remember. However, I have come prepared. I am about to share with you a pro tip for creating complex passwords that are easy to remember. The first thing you need to do is come up with a phrase or song lyrics or famous quote that has at least 8 words in it. Something that you will easily remember. As examples, here are a bunch of phrases that you could use (*supply the group with the laminated cards with the phrases on*). For each of the words, take the first letter. For each of those letters try to either convert it into a special character of number that could represent it and mix the case. Try matching these passwords to those phrases to help explain (*supply the passwords*).	4 mins
Conclusion	And as easily as that you have a reliable way of	2 mins

	remembering your complex passwords. No longer will you need to settle for poor, easily cracked passwords. No more will you struggle to remember your passwords. You can have strong, reliable passwords that contain a mixture of upper and lower case letters, special characters and numbers that is at least 8 characters long and contains no real words, without having to keep a logbook of all your secret login details for every account you have.	

Admin

In this task there is very little required. Perhaps a desk or some floor space to put the cards out on would help. The brute forcing activity can be done verbally but might hold more water if we could write it on a whiteboard. This is not a necessity but might be nice.

The laminated cards have already been produced. However, if another set is required then put a request into ReproGraphics. Use Appendix A as your source file and ask for laminating and cutting. Each cell should be a different card.

DDOS

Introduction

This workshop is all about how a DDOS works and why it's so effective and difficult to prevent. The workshop should focus on the fact that there are too many requests for the server to handle and that the requests come from more than one source. In this way, the students should get a small sense of how the world wide web works and just how fragile it is.

Script

Activity	Script	Timings
Intro	Hello Everybody and welcome to the DDOS segment. In this session I'm hoping to impart some small sense of how the World Wide Web works and how some people try to tear it down. To start off, does anyone know the difference between the Internet and the WWW (answer: the Internet is a series or interconnected networks, the WWW is a collection of Web pages that are hosted on the Internet)? How do you get a webpage to your device (answer: HTTP request goes to web server. HTTP response comes back, containing web page)?	2 mins
Demo	Let's get a nice easy visual representation of this system. I need one person to volunteer to be the webserver. I need another volunteer to be the device. To start out, the device will send a request to the server (direct the device to throw a coloured ball to the server in an underarm fashion). The server then needs to process that request. In this case, all it will do is drop it into the correct bucket. If there is no bucket to match the ball, drop it on the floor. (Direct the device to slowly start throwing balls from the bag to the server, one at a time. There is no need to do loads of these, maybe 5 should give them the right idea). And stop there. Was that easy? Now let's add one other device (Repeat the process but with two devices this time. Both throwing underarm and at their own pace). Now everybody get involved (This could get very boisterous. Please try to maintain the underarm rule to avoid injuries). And stop there, I think you have gotten the idea.	8 mins
Conclusion	So what happened (explore their observations. The key aspect to get out of this are that there were too many requests for the server to handle)? As you have correctly pointed out, there was way too much going on for the server to do anything correctly. In reality, when the server can't cope anymore and its queue is full, it simply	5 mins

	gives up and crashes. This is what is called a DDOS attack. People wishing to crash a website will gather together as many devices as possible and get them all to send requests to the server at the same time. This is sometimes enough to take the website offline. What could you do to prevent the DDOS (answer: not a lot. You need people to be able to send requests to your server? You could limit the number of requests that can come from each machine in a given period of time – e.g. only 3 requests in a 5 minute period – and you could add more servers so that more requests can be dealt with at the same time)? Hopefully with this exercise you now have a simple view of how the WWW operates and some of the vulnerabilities that you may face if you ever want to run your own websites one day.	

Admin

This is quite possibly the most boisterous activity since they will be throwing balls at each other. To that end it is quite important to stamp out poor behaviours in the interest of safety.

In terms of setup, this workstation needs three buckets that are labelled in such a way as to make it obvious what colour balls are supposed to go in which bucket.

Packetisation

Introduction

In this workshop the idea of packets is introduced. The fact that information does not travel in one great lump, but is in fact split into smaller 'packets' and transported with several restrictions is all that is expected to be taken away from this workshop.

Script

Activity	Script	Timings
intro	Moving data across a network is only possible by splitting the content into packets. Imagine if you were to buy a new house, you want to fill the house with new furniture, this would have to come in separate packages as it won't all fit in the same van. The same analogy can be applied to the Internet. If you want to transmit some content then it needs to fit on the wire/connection you are using.	2 mins
Task	So, you are going to help me get my YouTube page from the server, can I have a volunteer for the server please, to the client, can I have a volunteer for the client please. Now, the remainder of the team, you are going to be the network for me. Your job is, individually, to transfer as many of these packets (the balls) from the server to the client as you can. Only one person can be transmitting packets at any given time, so you will need to take it in turns, you must go around the turning point, and you must all take the same number of packets (balls) in one go. *Given a 30 second time, see how many balls the group can transmit from the server to the client. Expect this to be one of the lowest scores.*	5 mins
Extension 1	Now you have practiced transmitting packets along a single connection, how could we speed up the process? *Students should recognise that there was a limit on 1 person at a time, now we are looking for 2 people at a time (increased bandwidth).* That's brilliant, if we increase the number of packets we can send in one go, the bandwidth, we can speed up our link. Let's try the same exercise again and see how much of my video you can get to the client.	5 mins
Extension 2	So far we have tried running a single path, using packets to get content to the client. The Internet works slightly differently in that the path can change from packet to packet. Could I have some volunteers to be nodes on our network please?	5 mins

	Excellent, so, node 1 you are going to take all blue packets (blue balls), node 2 red packets (red balls) and node 3 all other packets. The server is going to sort the balls into coloured packets whilst I prepare you all. *Vary the length of each of the routes, one route can deliver straight to the client, one has to travel a little further and one has to take the full circuit. Any remaining students will form additional links in the chain. You should see from this exercise that we can get more content to the client if the server is quick enough at sending data out.*	

Admin

This session could get quite active and boisterous. Please be careful not to allow them to hurl the balls either at each other or at anyone else in the room.

Set up a space such that the students have space to run from one location to another, via certain 'nodes' or markers (use chairs or cones or whatever you can get your hands on).

TCP/UDP

Introduction

In this workshop we are showing two of the transport layer protocols that are widely used over IP networks: Transmission Control Protocol and User Datagram Protocol. The reason for this is to show the students just some of the ways in which Internet traffic is delivered and help explain why some things are more reliable than others when browsing the Internet

Script

Activity	Script	Timings
Intro	Hello and welcome to the Transport layer. In this session you will learn a little bit about some of the ways that data can be sent around the internet. We will be looking at two ways of sending data: the Trasmission Control Protocol (TCP) and the User Datagram Protocol (UDP). You don't need to remember these names unless you intend to go into this field as you get older. For now all you need to remember is that they are different methods of getting information from one machine to another over the internet.	2 mins
TCP	We will start by looking at TCP. This is a nice and reliable protocol. Each and every packet of data sent using this method is numbered so that they can be reassembled in the right order at the other end and if any are missing, the receiver can ask the sender to send another one. Let's see this in practice. I need two volunteers to play the sender and the receiver. The sender's job is to take these packets and send them one by one to the receiver. The receiver needs to catch them and reassemble the image on the table/floor/wherever. To make sure that the packets get there, use plenty of communication to ensure that you are both ready to make the pass	5 mins
UDP	Now in contrast we will look at UDP. This method is sometimes called the fire and forget method. In this method there is no two way communications to ensure that both parties are ready, there are no sequence numbers to help you reassemble the image. There is no guarantee that all of the packets will even arrive at their destination. To simulate this method I need two new volunteers to	5 mins

	play the sender and receiver. The sender will stand with their back to the receiver and throw packets over their shoulder towards the receiver. The receiver cannot move from where they are stood in order to catch the packets so the sender needs to be throwing in a reasonable direction. The sender can also keep firing packets at whatever rate they want. The receiver's job is to catch all of the packets and try to reassemble the image like a jigsaw puzzle.	
Conclusion	What were the advantages to the TCP method (the answers you are trying to steer the group towards are reliability, no loss of packets, easy of reconstruction on the other side)? What about the disadvantages (slow, lots of extra information used – i.e. a lot of overhead and bandwidth)? What was good about the UDP method (fast, lightweight in terms of overhead)? And the bad points (Un-reliable, lost packets)? Lots of information is sent over the internet. Some of which you would want to use TCP and some you would want UDP. For example, your web page might be sent using TCP because you need to make sure that you get all of the information. However, something that is real time like a telephone call could use UDP because you need the speed and you can tolerate the odd missing packet or two because you can still figure out what is being said.	3 mins

Admin

This workstation has the potential to get a little rowdy. To help alleviate this, only one person should be throwing and one person catching at any one time. This will not only help prevent the boisterousness but also improve the safety of the exercise.

Make sure that you have everything you need for this activity including the two large pictures that have been cut into smaller pieces (one picture is numbered on the back, the other is blank on the back). Some sort of receptacle like a box or bucket could be used for the 'client' to catch the responses from the 'server' but they can just catch the packets with their hands.

Appendices

Appendix A

King George commands and we obey, over the hills and far away
When in doubt, check them out, Go Compare

One small step for man, one giant leap for man kind
To be or not to be, that is the question
Mary had a little lamb who’s fleece was white as snow

What's the story, Balimory? Wouldn't you like to know?
You spin me right round baby, right round

Imagination is more important than knowledge – Albert Einstein
Do you wanna build a snowman?

Twinkle Twinkle Little star, how I wonder what you are.
Oooooooh Who lives in a pineapple under the sea?
Luke. I am your father.

To infinity and beyond.
Pikachu
Is it too late now to say sorry?
I was running through the 6 with my woe.
kGc&w0O7f4Fa

W!dc70gC
O5sFm19l4MK
T3o!7btitQ
Mh4Llw5Wwa$
w7$3Wyl2K?
Y$mrR3Rr
I!mi7K4E
dUw8@$M

T7l*h1Ww^r
0o)WL!aP4_tC?
L>!aURfa4.
2inF!NiT&8yND
pE3K@(Ho0
l!2LnT$5
1wR7t$+mW