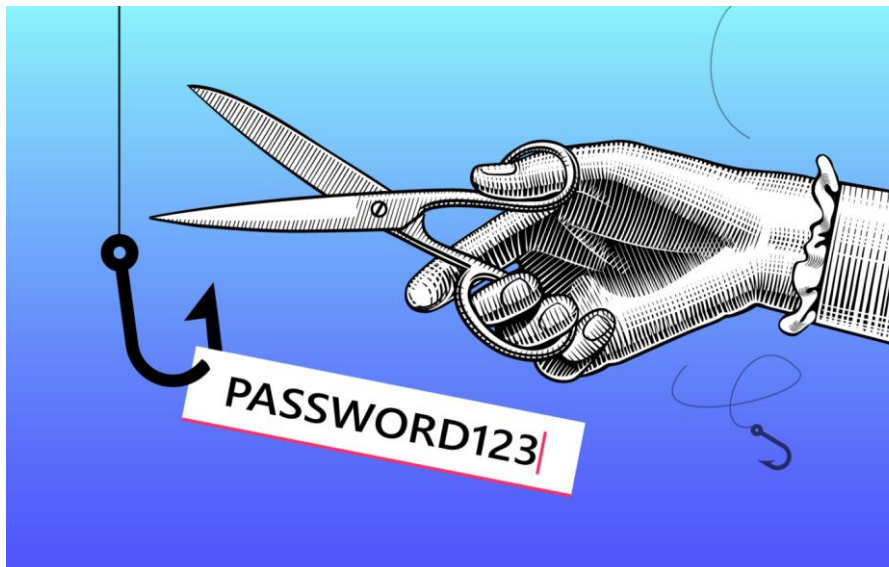


Project Security

Project Security IP Paper 22-23

“Better safe than sorry”



Nils van Witzenburg | 500849196 | IC202

11-11-2022

Versie 1.0



Inhoudsopgave

Inleiding.....	2
Problematiek	2
Aanpak en oplossing	3
Ontwerp en implementatie.....	4
Conclusie en advies.....	7
Bronnen.....	8
Bijlage 1 – Poster	9
Bijlage 2 – Poster	10
Bijlage 3 – Code van script	11
Bijlage 4 – Demo	12
Logboek.....	13
Docent feedback	13
Ontvangen en gegeven feedback	13
Planning.....	14
Deadlines	14
Sprint 1	14
Sprint 2	15
Sprint 3	16
Retrospective Sprint 1	17
Retrospective Sprint 2	17
Retrospective sprint 3	17
Methoden.....	18
Individuele reflectie	19

Inleiding

Bedrijf Y heeft net zijn deuren geopend en haar IT systeem ingericht. Het bedrijf heeft een mailserver die gebruikt wordt voor de klantenservice. Op de website kan een form worden ingevuld dat wordt doorgestuurd naar de mailserver. Gelukkig gaat het bedrijf onderzoek doen naar potentiële phishing-aanvallen. Ik, als ICT'er bij het bedrijf, heb de opdracht gekregen om de mailserver te beveiligen tegen deze aanvallen.

Mijn doel is om de hoeveelheid phishing e-mails te verminderen, die de medewerkers binnen krijgen via de form op de website. Vooral andere bedrijven en klanten vullen deze form in. Bij het invullen van de form moet degene hun eigen e-mail meegeven, waar zij een antwoord ontvangen in maximaal 5 werkdagen. Het doel van deze paper is om bedrijven op ideeën te brengen hoe je phishing-aanvallen tegen kan gaan, kleine of grote ICT afdeling.

Problematiek

Het bedrijf heeft mij dus gevraagd om onderzoek te doen naar het tegengaan van phishing-aanvallen.

Volgens Zscaler (2020) is door Covid-19 phishing, malicious websites en malware aanvallen, gerelateerd aan Covid, met 30.000% gestegen. Phishing-aanvallen zijn gestegen met wel liefst 667% en volgens APWG (2021) bereikte het aantal phishing-aanvallen een recordhoogte in 2021. Bedrijven en goede doelen zijn de meest voorkomende slachtoffers volgens de regering van het Verenigd Koninkrijk, GOV.uk (2022). Het blijkt dat Phishing-aanvallen de populairste vorm van cybercrime zijn. Daarom is het als beginnend bedrijf een van de belangrijkste uitdagingen om zoveel mogelijk van deze aanvallen te voorkomen en tegen te gaan.

Mijn probleemstelling is dus: Wat is voor bedrijf Y de beste maatregel om phishing-aanvallen tegen te gaan op korte termijn?

Aanpak en oplossing

Om te beginnen heb ik onderzoek gedaan naar welke mogelijke oplossingen er zijn tegen phishing-aanvallen.

Ik heb gekeken naar A.I. Based Protection. Dit is een goede oplossing volgens Santeri Kangas (2022), maar dit wordt al snel erg te duur en lastig om te onderhouden. Ik zou een eigen A.I kunnen programmeren, maar dit zal teveel tijd gaan kosten en dus op korte termijn minder veiligheid bieden.

Er worden veel cursussen aangeboden in het herkennen van (gerichte) phishing. Phishing e-mails worden met de dag 'echter' en het vergt energie en kosten om up-to-date te blijven. Verder zijn human errors nooit uit te sluiten. Maar dit is wel een goede maatregel om toe te voegen als het bedrijf meer personeel in dienst heeft.

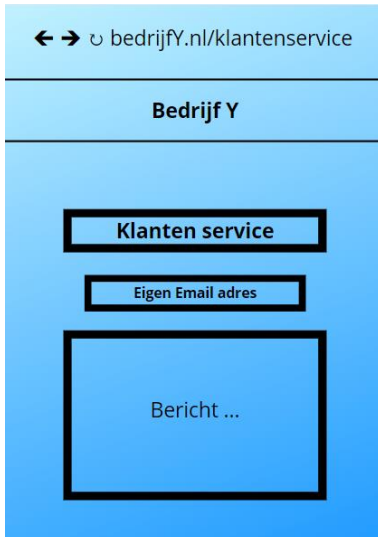
Tenslotte ben ik zelf gaan denken over mogelijke alternatieve oplossingen. In het verleden heb ik geprogrammeerd in Python. Met mijn Python kennis zou ik een script kunnen maken die phishing e-mails kan detecteren op basis van een aantal veelvoorkomende eigenschappen. Ik heb daarbij gekozen om een Python script te maken, omdat het redelijk simpel te implementeren is, het efficiënt is en daarmee relatief goedkoop op korte termijn zekerheid voor het bedrijf wordt geboden.

Het script zal phishings emails moeten kunnen detecteren op basis van email adres en link(s). Er zal een tekst bestand worden aangemaakt, waar email adressen worden ingezet die een red flag geven. Deze red flag geeft aan dat dit een phishing email kan zijn. Verder wordt er gecheckt op interpunctie bij de domein naam van het email adres. Omdat de meeste e-mails binnen komen van klanten, wordt er gebruik gemaakt van populaire domain namen. Daarom zal bijvoorbeeld 'go-ogle' worden geflagged als verdacht. Als laatste zal er gekeken worden of er een link staat in de email. Het zal ongebruikelijk zijn als een klant een link stuurt waar een medewerker op moet gaan klikken. Daarom zal het script ook signaleren als er een link staat in de email. Verder roept het script een derde partij aan, die controleert of deze link malware, spam of phishing bevat.

Het risico van een script is dat het de tekst in email niet goed kan lezen. Er wordt gebruik gemaakt van een Python library die de tekst leest en omzet in een string en dit wordt in de regel niet in 100% van de gevallen goed gedaan. Verder kan de medewerker de flags negeren en alsnog op een link klikken. Omdat er gebruik gemaakt wordt van een derde partij, heeft het bedrijf geen controle over de checks op de gevonden links.

Ontwerp en implementatie

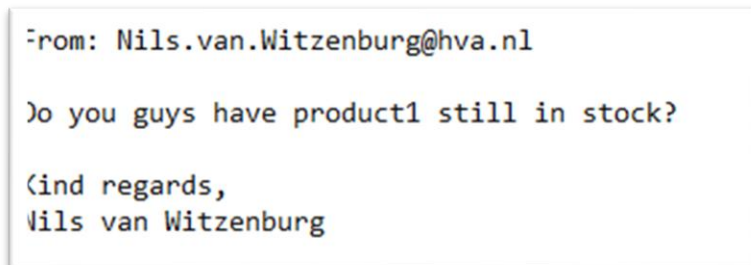
Het bedrijf heeft een website. Op deze website staat een klantenservice form, zie afbeelding 1 hieronder.



The screenshot shows a web browser address bar with 'bedrijfY.nl/klantenservice'. Below the address bar is a header 'Bedrijf Y'. The main content area is a light blue box containing three input fields: 'Klanten service', 'Eigen Email adres', and a larger text area labeled 'Bericht ...'.

Afbeelding 1 – Klantenservice form

Op deze form zal de klant zijn gegevens invullen. Het bedrijf controleert vervolgens of het een geldige email adres is. Als het email adres geldig is, zal de ingevulde form worden omgezet naar email en gestuurd worden naar de mailserver van het bedrijf. De medewerker zal deze email krijgen in zijn of haar mailbox, zoals in het voorbeeld hieronder.



The screenshot shows an email interface with the following content:

From: Nils.van.Witzenburg@hva.nl

Do you guys have product1 still in stock?

Kind regards,
Nils van Witzenburg

Afbeelding 2 – Email voorbeeld

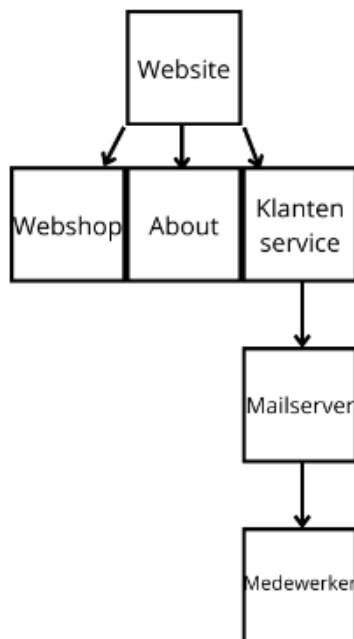
Het Python script zal uiteindelijk automatisch werken op de mailserver. Omdat ik tijdens dit project niet genoeg tijd heb om een mailserver te maken, heb ik besloten om een script te maken die handmatig gestart moet worden. In mijn Github¹ staat een beschrijving hoe het script gedraaid kan worden, verder is de code te zien in Bijlage 3. Het Python script controleert de interpunctie, email adres en links. Er wordt gebruik gemaakt van Python3, OpenCV-python en Python-tesseract.

Om te controleren of het een 'veilige' link is, wordt er gebruikt van een derde partij 'IPQualityScore'. Het heeft een fraud-prevention tool. Het detecteert 'bad actors' en blokkeert bots met gebruik van reputatiegegevens en gebruikersvalidatie. Ik roep de website API op en vervolgens geeft de website een JSON terug met data. Deze data bevat of de gecontroleerde website spam, malware of phishing kenmerken bevat.

¹ https://github.com/nilsvw/IP_ProjectSecurity

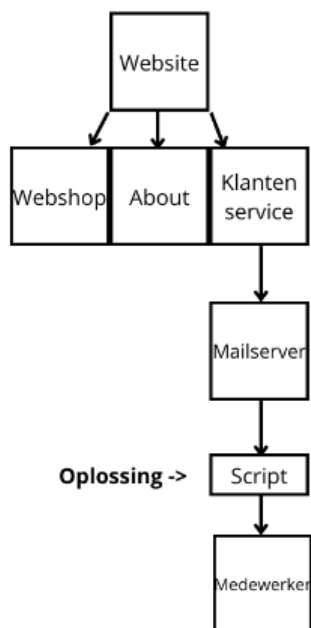
De script zal vragen om een deel tekst of een plaatje te scannen. Als de gebruiker gekozen heeft om een plaatje te scannen, zal het OpenCV en Pytesseract aanroepen.

Voor dat ik begon met het bedenken van een oplossing tegen phishing attacks, zag de infrastructuur van het bedrijf er zo uit.



Afbeelding 3 – Before

In afbeelding 4 staat waar mijn oplossing geïmplementeerd wordt.



Afbeelding 4 – After

Bij elke scan van een plaatje, wordt er een log bestand aangemaakt. In het log bestand staat de tijd van de scan, flags, checks en de volledig gescande tekst. De medewerker die de scan heeft uitgevoerd, kan de log bestand sturen naar mij als er enige twijfel is. De medewerker kan zelf ook de scan resultaten lezen uit de output van de terminal. Als de medewerker een flag ziet, moet hij of zij het direct melden bij mij.

Het script heeft een aantal zwakke plekken. Als er twee links staan in de e-mail, crashed het script. Dit komt door de manier waarop het script voor links checkt. Dit kan opgelost worden door de code te herschrijven zodat het de tekst blijft scannen voor links tot aan het einde. Verder kan het script verborgen links onder bijvoorbeeld een afbeelding, knop of hyperlink niet vinden. Dit kan opgelost worden door in de pakketjes van een email te kijken, nu wordt alleen gekeken naar het plaatje van de e-mail. Als laatste zwakke punt is dat de Python Tesseract library af en toe de tekst niet goed scanned. Dit kan ook opgelost worden door in de pakketjes te kijken van de e-mail.



Conclusie en advies

Dus het script scant het email adres op interpunctie en de blacklist. Verder als er een link staat in de email, wordt dit gedetecteerd. Vervolgens wordt deze link gescand met behulp van een derde partij.

Advies is om het script niet volledig te vertrouwen, omdat het af en toe de tekst niet goed kan scannen. Script heeft nog zwakke punten. Ook moet je als medewerker het log bestand sturen naar ICT medewerker als je iets niet vertrouwd. Verder is het handig voor het bedrijf om een aantal cursussen over phishing-aanvallen te kopen. Organiseer een dag waarin iedere medewerker die werkt met klantenservice moet komen. Hier wordt uitgelegd wat het script doet, hoe je het kan gebruiken en dat het alleen een hulpmiddel is.

Uiteindelijk wil ik mijn script automatisch laten draaien op de mailserver. Het script zal in de pakketjes zelf kijken. Op deze manier lost ik het probleem op dat het script niet links kan vinden die onder een afbeelding, knop of hyperlink zitten.

Met het script verminder ik veel succesvolle phishing aanvallen. Medewerkers zullen minder snel in phishing emails trappen, wat het bedrijf een stuk veiliger maakt. Om even terug te komen naar mijn probleemstelling: Wat is voor bedrijf Y de beste maatregel om phishing-aanvallen tegen te gaan op korte termijn? De beste maatregel op korte termijn is dus een Python script die de emails controleert. Op lange termijn zal het Python script aangepast worden, zodat het automatisch gedraaid wordt op de mailserver, en er cursussen worden gegeven over phishing-aanvallen.

Bronnen

APWG. (2022, February 23). PHISHING ACTIVITY TRENDS REPORT. *Apwg.Org*.

https://docs.apwg.org/reports/apwg_trends_report_q4_2021.pdf

Cyber Security Breaches Survey 2022. (2022, October 27). GOV.UK.

<https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2022/cyber-security-breaches-survey-2022>

DataFlair. (2019). Pros and Cons of Artificial Intelligence – A Threat or a Blessing? *Data-Flair.Training*.

<https://data-flair.training/blogs/artificial-intelligence-advantages-disadvantages/>

Desai, D. (2021, February 24). *30,000 Percent Increase in COVID-19-Themed Attacks*. Zscaler.

<https://www.zscaler.com/blogs/security-research/30000-percent-increase-covid-19-themed-attacks>

Witzenburg, van, N. (2022, November 7). IP_ProjectSecurity. Github.

https://github.com/nilsvw/IP_ProjectSecurity

Why AI is the key to developing cutting-edge cybersecurity. (2022, July 25). World Economic Forum.

<https://www.weforum.org/agenda/2022/07/why-ai-is-the-key-to-cutting-edge-cybersecurity/>

Bijlage 1 – Poster

Versie 1 voor feedback

HOW TO PREVENT PHISHING ATTACKS

Aanvallen met **667%** gestegen
Populairste vorm van Cybercrime

Nils van Witzenburg
Nils.van.Witzenburg@hva.nl
500849196
IC202



SOLUTION:

- Klein programma dat snel en simpel is



Python Script dat E-mails controleert



"I'm so happy I used this script instead of something expensive!"

Bijlage 2 – Poster

Versie 2 na feedback

HOW TO PREVENT PHISHING ATTACKS

Aanvallen met **667%** gestegen
Populairste vorm van Cybercrime

Nils van Witzenburg
Nils.van.Witzenburg@hva.nl
500849196
IC202



Python Script dat E-mails controleert

SOLUTION:

- Klein programma dat snel en simpel is



"I'm so happy I used this script instead of something expensive!"



Bijlage 3 – Code van script

https://github.com/nilsvw/IP_ProjectSecurity



Bijlage 4 – Demo

Video komt hier van demo

Link =

Embed:

Logboek

Docent feedback

06/10/2022

Na het geven van mijn pitch keek de docent uit naar waar mijn zwakke punten zitten in het script.

02/11/2022

“Is het belangrijk voor de gebruiker om te weten waar de scanner wel/niet op scant zodat die niet per ongeluk denkt 100% safe te zijn? Soort disclaimer? En afsluiten met twee adviezen. Advies om toch voorzichtig te zijn of advies om dit de moeite waard lijkt om te melden bij personen in de organisatie die over information security gaan of dat ze moeten controleren of ze het goede aan het scannen zijn. Dat soort zaken. Dan bij de alert: specifiek wat er waar gevonden is zodat de security persoon goed inzicht heeft? Soort rapportje/pdf misschien?”

Ik ben de volgende dag aan de slag gegaan om deze feedback te verwerken.

Ontvangen en gegeven feedback

Poster Noa, structuur verbeteren door alle tekst links en plaatjes rechts bijv. of stukjes tekst hoofdstuknummers geven

Robin, sommige tekst iets vergroten, maar erg duidelijke poster

Goede duidelijke stem tijdens presentatie

Goed dat ik presenteerde over wat niet op mijn poster staat

Duidelijk praten

Fijn dat er buiten de poster gepraat wordt

Aangeven waar de phishing attacks voorkomen

Mooie poster met een logische volgorde

Komische element is een leuke toevoeging

Planning

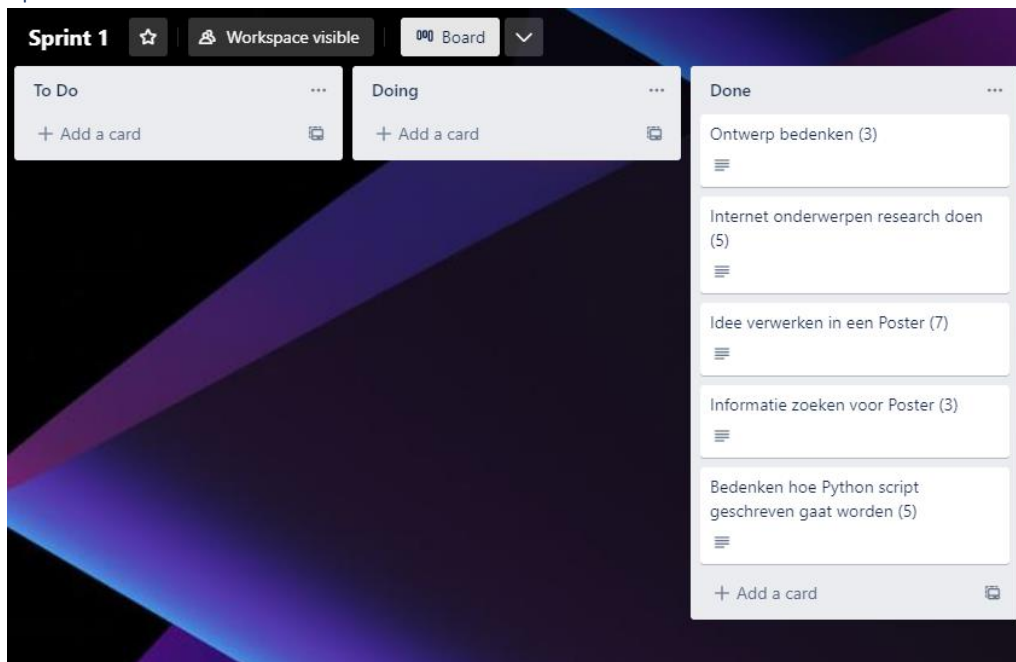
Ik heb ervoor gekozen om Trello te gebruiken als mijn planner. Hier kan ik duidelijk voor mijzelf weergeven wat ik nog moet doen, waar ik mee bezig ben en wat ik heb afgerond. De planning is verdeeld in drie sprints. In het hoofdstuk 'Werkwijze' wordt uitgelegd waarom. In de afbeelding per sprint staan alle taken.

Deadlines

06/10/2022 Pitch Poster

18/11/2022 Inleveren Individueel Product

Sprint 1



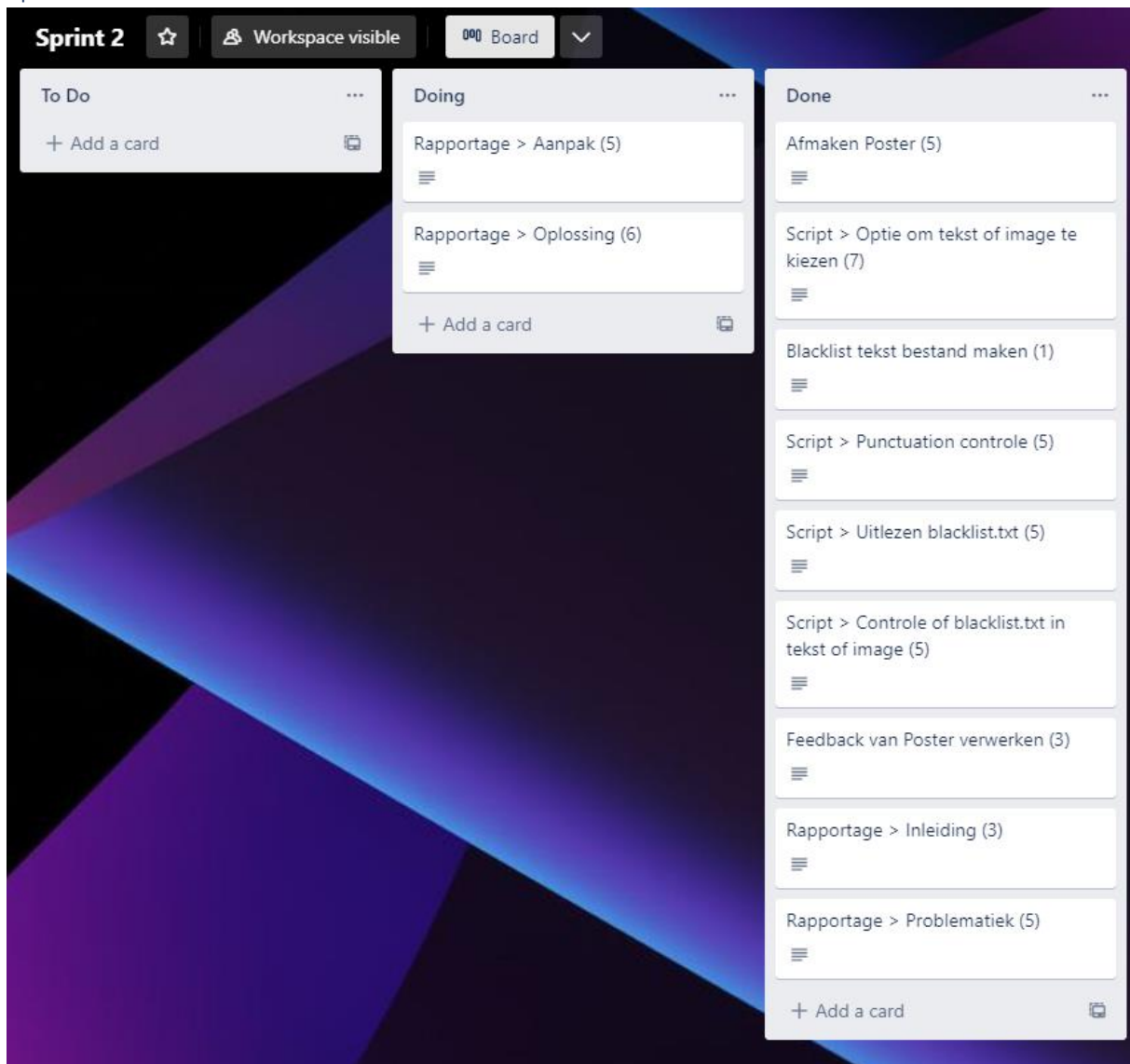
06/09 en 07/09 ben ik begonnen met het bedenken van Cyber Security problemen. Ik heb op internet opgezocht of er veel te vinden is over het onderwerp en of het wel mogelijk is om gekozen probleem op te lossen.

09/09 ben ik begonnen met het maken van mijn poster voor de Pitch. Hier heb ik mijn ideevorming vast gesteld: Script die phishing emails kan detecteren.

11/09 heb ik de studiehandleiding doorgelezen. Bekeken wat er allemaal in de poster moet.

16/09 ben ik onderzoek gaan doen over mijn onderwerp. Dit is nodig voor de Rapportage en Poster. Bronnen genoteerd in Rapportage en mijn poster ingericht. Ook ben ik begonnen met het schrijven van mijn script. Dit is nog heel erg de basis en het kan alleen nog maar de keuze van de gebruiken vragen, Image of Tekst scannen.

Sprint 2



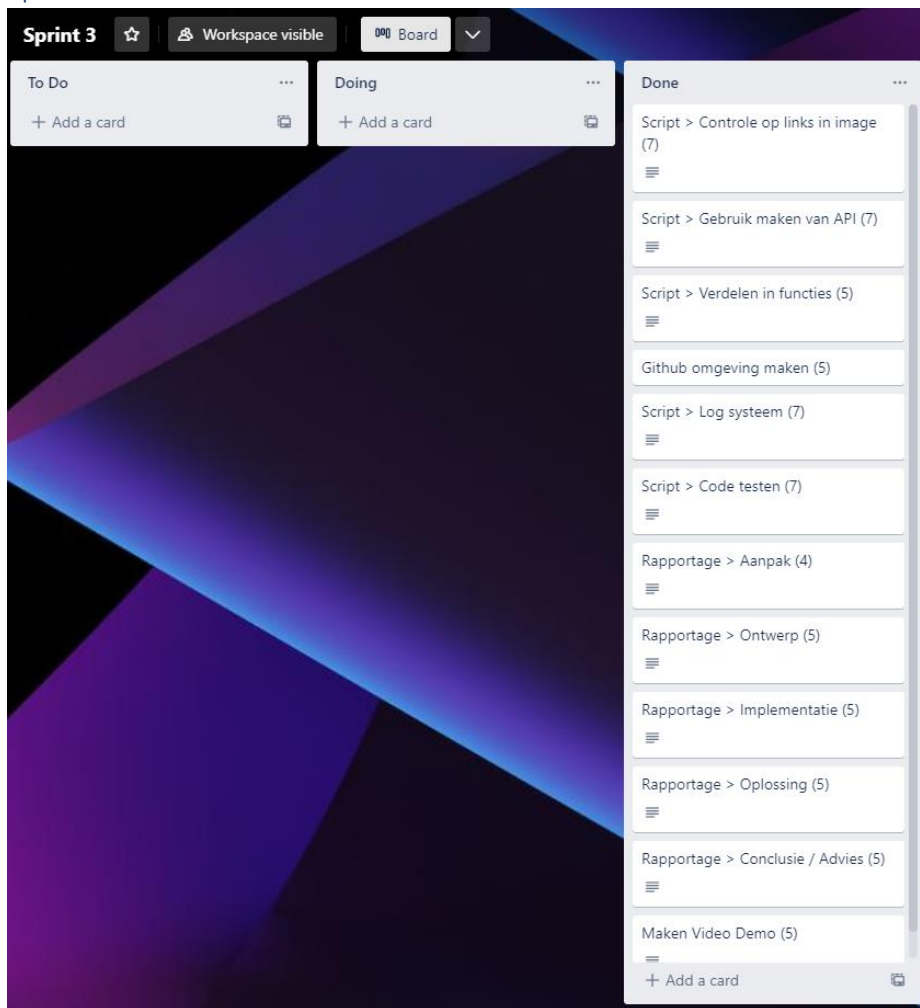
19/09 Script, afmaken optie om tekst of image te scannen. Ook poster afgemaakt voor de Pitch.

22/09 Script, tekst bestand gemaakt genaamd 'Blacklist.txt'. Code geschreven om tekst bestand uit te lezen. Rapportage inleiding en problematiek gemaakt.

06/10 Pitch gegeven met poster. De feedback ontvangen van medestudenten. Ben dezelfde dag feedback gaan toepassen op de poster. Begonnen met schrijven van Aanpak en Oplossing. Dit zijn momenteel alleen steek worden, omdat het lastig is om nu al alles 100% te weten.

07/10 Implementeren en schrijven code om tekst uit een plaatje te lezen en op slaan. Toevoeging dat bij beide opties (tekst en image), er gecontroleerd wordt of er enige interpunctie in email-domein staat.

Sprint 3



09/10 Maken code dat controleert of er een link staat in de tekst van de image.

14/10 API gebruiken die de link gebruikt uit de image. Dit betekent het opvragen van een API-key om de API te kunnen gebruiken. Daarna moet de code een link vinden, opschonen en doorgeven aan de API.

25/10 Code in verschillende functies zetten zodat de code efficiënter en overzichtelijker is. Verder heb ik vandaag de code getest met behulp van verschillende situaties.

27/10 Gewerkt aan de Rapportage. Aanpak en Oplossing afgemaakt. Verder heb ik gewerkt aan Ontwerp en implementatie. Dit 90% afgemaakt.

28/10 Code in Github omgeving zetten. Test afbeeldingen, script en blacklist.txt toegevoegd. Verder een README bestand gemaakt waarin staat hoe je zelf thuis het script kan gebruiken.

03/11 Alle bugs uit het script gehaald. Log systeem gemaakt.

04/11 Afmaken van Aanpak en Oplossing in Rapportage. Ook Ontwerp en Implementatie afgemaakt.

11/11 Maken conclusie in Rapportage en maken van DEMO video. Hierbij sluit ik ook sprint 3 af.

Retrospective Sprint 1

Aan het einde van een sprint ga ik reflecteren op mijn proces en werkwijze. Tijdens sprint 1 heb ik vooral onderzoekend werk gedaan. Dit is van belang omdat ik kan goed een beeld krijg van wat ik wil gaan maken. Ik ben tevreden over hoe het gegaan is tijdens deze sprint. Ik heb al mijn taken kunnen afmaken. Bij een volgend project ben ik van plan om meer taken te doen. Ik heb vooruit gekeken over wat ik allemaal nog moet doen, en zie dat ik meer taken moet doen tijdens sprint 2 en 3.

Retrospective Sprint 2

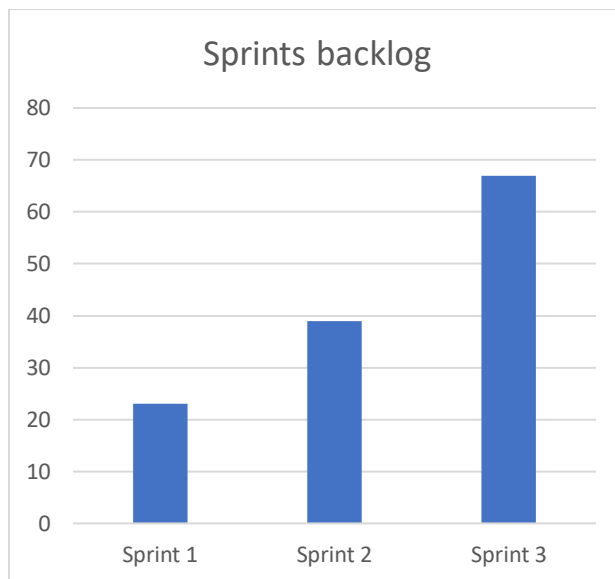
Tijdens sprint 2 ben ik hard te werk gegaan met het coderen van mijn script. Ik had gepland om wat meer aan de rapportage te werken, maar ik kwam er achter dat dit nog niet kon. Dit kon pas wanneer ik mijn script bijna af heb. Dit is geen probleem, omdat ik de twee taken meeneem naar sprint 3.

Retrospective sprint 3

In sprint 3 heb ik de Rapportage en Script af kunnen maken. Omdat ik goed had ingeschat hoeveel tijd ik ongeveer nodig had om taken af te ronden. ...

Sprints backlog

Op basis van de punten die ik elke userstory heb gegeven, is er een sprints backlog gemaakt. Het geeft visuele inzicht over de voortgang per sprint.



Wat opvalt is dat het bijna een lineaire stijging is. Het is erg lastig om al met de documentatie van het product te beginnen als ik nog niet een volledig product heb gemaakt. Daarom zijn veel punten in sprint 3 documentatie. In sprint 1 heb ik onderzoek gedaan en een poster gemaakt van mijn gekozen onderwerp. In sprint 2 ben ik begonnen aan het script, wat meer tijd kost, wat ook te zien is. In sprint 1 ben ik gaan plannen voor sprint 2 en 3. Wat ik merkte is dat sprint 2 en 3 mij veel meer tijd gingen kosten, daarom heb ik hier rekening mee gehouden.



Methoden

Gebruik gemaakt van scrum

Scrum eigenlijk voor team project bedoelt, maar voor individueel omdat ik mij dan forceer om te reflecteren over mijn werk. Dit helpt mij. Ook goed overzicht houden

Product owner en scrum master ben ik zelf, omdat individueel product

Product backlog is Trello

User stories verwerkt in Trello bij elk kaartje toegevoegd als beschrijving

3 sprint backlogs -> 3 sprints doen

Retrospective aan eind van sprint.

Websites ect die zijn gebruikt en waarom

Internet

Onderzoekmethodes ...



Individuele reflectie