Kauno technologijos universitetas

Informatikos fakultetas

Modulis „Tiriamasis projektas 1“
Projektas: Mašininį mokymąsi naudojantis Google Chrome naršyklės įskiepis, vertinantis interneto svetainių saugumą pagal URL adresą

Projektavimo metodologijos ir technologijų analizė

IFM 9/2 gr. Liudas Kazalupskis

Studentas / Studentė

Prof. Robertas Damaševičius
Projekto vadovas

Lekt. Virginija Limanauskienė

Dėstytoja

Kaunas, 2019

Turinys

Lentelių sąrašas

Santrumpų ir terminų sąrašas

Santrumpos:

URL (angl. Uniform Resource Locator) – šnekamojoje kalboje vadinamas tiesiog svetainės adresas, tai nuoroda į internetinius resursus, svetaines.
APWG (Anti-Phishing Working Group) (liet. kovos su sukčiavimu darbo grupė) – tarptautinis konsorciumas, kurio tikslas kovoti su kibernetiniais nusikaltimais visos industrijos sferose.
SSL (Secure Sockets Layer) – protokolas užtikrinantis saugu ir užšifruotą bendravimą tikle.

Terminai:

Fišingas (angl. Phishing) - sukčiavimo forma, skirta išvilioti konfidencialius duomenis, naudojant internetinius adresus, panašius į iš tikro egzistuojančios institucijos adresą.

Įvadas

Dokumentas yra Programų sistemų inžinerijos magistrantūros disciplinos „Tiriamasis projektas 1“ ataskaita. Dokumento paskirtis apibūdinti tyrimo tikslus, apibendrinti atliktą literatūros analizę, pasirengti projekto reikalavimų specifikavimui, projektavimui, susipažinti su užsakymo taikymo sritimi, pasauliniais pasiekimais taikomojoje srityje.

Fišingas yra vienas kibernetinių nusikaltėlių vagiliavimo būdas. Dažniausiai per el. laiškus, žinutes siunčiami laiškais, kuriuose apsimetama teisėtais vartotojais, tokiais kaip draudėjai, paslaugų teikėjai ar su darbu susijusiais asmenimis ir prašoma prisijungti per jų svetainę ir tai padarius jūs prarandate savo duomenis ar pinigus. Tokios atakos dažnai būna nukreiptos į didelio kapitalo įmonėse dirbančius žmonės ir taip kėsinamasi į konfidencialią įmonės informaciją.

Šiuos puolimus dažnai įmanoma identifikuoti pagal URL adresą. Vienas iš puolimo tipų yra URL adrese raidės sukeistos vietomis. Vienas iš adresų pavyzdžių galėtų būti „aleixpress.com“. Iš pirmo žvilgsnio šis adresas gali būti panašus į elektroninės parduotuvės „Aliexpress“ adresą, parastai atsidarius ir turinys būna labai panašus, tačiau iš tikrųjų taip nėra, jeigu žmogus prisijungtų, jo duomenys greičiausiai atitektu kibernetiniam nusikaltėliui.

Įskiepis, kuris gebėtų identifikuoti tokio tipo internetines svetaines pagal jų URL adresą padėtų žmonės apsaugoti asmeninius duomenis, o įmonėms įvairią konfidencialią informaciją nuo itin dažnai pasitaikančių kenkėjiškų interneto svetainių.

Atsižvelgiant į tai, kad vis daugiau sutarčių, sąskaitų ar kitokių dokumentų yra saugomi kompiuteriuose ar internetinėje erdvėje, fišingo tipo išpuolių skaičius tik augs ir jie tobulės, dėl to įrankis, kuris padėtų neįkliūti bus aktualus ir ateityje.

Raktiniai žodžiai: Phishing, Phishing detection, Phishing detection using artificial inteligence, Phishing detection using machine learning, Phishing URLs

1. Tikslas

Šio darbo tikslas yra pasitelkus mašininį mokymą apmokyti modelį naudojant viešai prieinamus duomenis su kenkėjiškais ir tinkamais URL adresais, kuris gebėtų didesniu nei 97% tikslumu atpažinti kenkėjiškas interneto svetaines.
Realizuojant išorinę sąsają reikės sukurti naršyklės įskiepį, kuris identifikuotų kenkėjiškas internetines svetaines prieš vartotojui jas pasiekiant iš URL adreso naudodamas anksčiau minėtą modelį.

2. Fišingo nusikaltimų tendencijos analizė

Remiantis APWG ataskaitos duomenimis [5] , fišingo išpuolių kiekis 2019 metų trečiajame ketvirtyje pakilo iki tokio aukšto lygio, koks buvo pasiektas tik vėlyvuosiuose 2016 metais. Iš viso buvo APWG aptiko 277,693 fišingo tinklapius (antrajame ketvirtyje buvo aptikta tik 182,465 [6]) ir gavo 122,359 pranešimus apie kenkėjiškus elektroninius laiškus. Taip pat net 68% išpuoliams skirtų svetainių turėjo SSL protokolą, palyginus su praeitu ketvirčiu šis skaičius siekė tik 54%, o praėjusiais metais nesiekė 50%.

3. Fišingo atpažinimo pagal URL adresą esamų sprendimų analizė

Ko gero svarbiausias šio projekto uždavinys yra kuo tiksliau atpažinti kenkėjiškus URL adresus. Atpažinimas pagal adresą gali būti labai tikslus, dėl kad neįmanoma sukompromituoti legalaus adreso nepakeičiant jo struktūros [2] . Sukompromituoti URL adresą galima labai įvairiais būdais, tokiais kaip papildomo simbolio pridėjimas, domeno pabaigos pakeitimas pavyzdžiui iš „.com“ į „.org“. Naudojant sukauptus duomenis apie fišing svetainių adresus, taip pat surinkus legalių svetainių URL, tinkamai išrinkus bruožus ir sukūrus taisykles, kurias pritaikius galima atskirti kenkėjiškus ir tinkamus adresus, ir panaudojus mašininį mokymąsi įmanoma apsimokyti modelį, kurio tikslumas yra labai arti 100%. Sprendimų, kokius bruožus geriau naudoti ir kokia taisyklės pritaikyti, yra daug ir labai įvairių, bet ne visi jie yra vienodai efektyvus sprendžiant šią užduotį. Tačiau atliekant tyrimus išrinkti tinkamus ir kuo įvairesnius duomenis, nes kitu atveju tyrimo rezultatai gali labai skirtis nuo tikslumo realiuose situacijose. Pavyzdžiui vienoje publikacijoje, tiriančioje fišingo svetainių atpažinimą buvo panaudoti JRIP ir PART algoritmai pritaikant tas pačias taisykles. Tikslesni rezultatai šio tyrimo metu buvo naudojant PRAT algoritmą (99.08%) [1] . Kitoje publikacijoje pagal tas pačias taisyklės ir su tokias pačiais duomenimis buvo patikrinti 7 skirtingi klasifikavimo algoritmai, o tiksliausias buvo „Random Forest“ algoritmas, kurio tikslumas buvo 97.9%, o nuo jo labai nedaug atsiliko „Random tree“ algoritmas su 97,8% tikslumu, tačiau jis buvo beveik 20 kartų greitesnis [13] . Didelė dalis straipsnių apie fišingo adresų identifikavimą pagal URL adresą buvo klasifikavimo tipo mašininio mokymosi algoritmų panaudojimas ir palyginimas. Tokio tipo algoritmų yra nemažai ir dažniausiai skirtingose srityse geriausia rezultatą parodo skirtingi algoritmai, galima sakyti, jog nėra vieno – tinkamo viskam. Šioje srityje geriausią rezultatą rodė, kaip ir ankščiau minėtos publikacijos atveju - „Random Forest“ algoritmas. Dauguma atveju atrinkti bruožai nebuvo vienodi ir buvo pasiekti tokie rezultai: 99.7% [8] , 89.9% [9] , 97.2% [10] , 98.46% [12] , 95% [19] , 98% [20] , 94.26% [21] , 97.98% [22] . Žinoma buvo ir tyrimu, kur puikius rezultatus pademonstravo kiti algoritmai: „SMO“ algoritmas 95.39% [4] , „Novel“ metodu paremtas rekurentinis tinklas 98.7% [7] , „J48“ 93% [12] , „PSO“ apmokytas rekurentinis tinklas 99.9% [14] , „SVM“ daugiau nei 90% [16] , „Apriori“ 93%, taip pat vieno tyrimo metu buvo panaudotas ir HTML statinis tyrimas, tačiau tikslumas nebuvo žymiai didesnis (98.6%) [18] .Buvo ir šiek tiek kitoks būdas identifikuoti kenkėjiškus URL adresus – naudojant perdiskretizavimą (angl. oversampling). Šiuo būdų taip pat reikėjo išrinkti taisykles būdingas fišingo svetainių adresams ir buvo sukurta daug skirtingų URL adresų, su kurias vėliau buvo lyginami adresai, kuriuos norima patikrinti ar jie yra legalių svetainių [3] . Taip pat buvo atliktas aplinkos efektyvumo atliekant URL adreso analizę tyrimas, kuris įrodė, jog efektyviau analizę daryti debesų technologijų aplinkoje negu nepaskirstytoje sistemoje.

4. Egzistuojančių rinkoje, programų palyginimas

Rinkoje pavyko surasti tris įskiepius gebančius apsisaugoti nuo fišing išpuolių, tačiau visi jie yra skirtingi ir tarpusavyje, ir nuo kuriamo įskiepio.

4.1. Kuriamas įskiepis

Kuriamas įskiepis tik pagal URL adresą su ~3% paklaida identifikuotų kenkėjiškas interneto svetaines. Šis įrankis pirmiausiai patikrintų juodąjį ir baltąjį sąrašą. Jeigu puslapis būtų juodąjame sąraše vartotojui būtų apie tai pranešta ir jis pats turėtų nuspręsti ar pavojus tikras. Jeigu yra baltąjame sąraše – vartotojas automatiškai praleidžiamas toliau. Jeigu URL adreso nėra nei viename sąraše jis yra tikrinamas mašininio mokymosi modelio, kuris yra apmokytas pagal duomenis, surinktus iš skirtingų šaltinių, apie kenkėjiškas ir tinkamas svetaines. Jeigu URL adresas identifikuojamas kaip kenkėjiškas – vartotojas turi patvirtinti ar tikrai kenkėjiškas ir jeigu taip adresas įtraukiamas į juodąjį sąrašą ir ateityje nebetikrinamas.

4.2. Netcraft Extension

Šis įskiepis turi daugiausiai funkcionalumų skirtų apsisaugoti nuo kibernetinių nusikaltėlių ir vienas jų yra apsauga pagal URL adresą. Kenkėjiškų URL adresų atpažinimas yra paremtas juodojo sąrašo principu. Tai reiškia, kad žmogus aptikęs kenkėjišką puslapį, gali apie jį pranešti ir jis bus užblokuotas visiems šio įskiepio vartotojams. Taip pat yra įtraukiami kenkėjiškų puslapių adresai, kurie yra gauti iš trečiųjų šalių.

Taip pats šis įskiepis turi tokias funkcijas kaip kenkėjiško „JavaScript“ aptikimas, detali puslapio ataskaita, saugaus užšifravimo tikrinimas, SSL sertifikatų tikrinimas ir apsauga nuo „XSS“ išpuolių.

Šio įrankio trūkumas lyginant su kuriamu įskiepio – jis negali atpažinti naujai sukurtų adresų, kol jie nepridedami į juodąjį sąrašą.

4.3. PhishDetector

Įskiepis skirtas apsisaugoti nuo fišing puolimų elektroninio banko svetainėse. Šis įrankis identifikuoja elektroninės bankininkystės puslapio turinį ir identifikuoja kenkėjiškus puslapius su nuline klaidingai neigiama paklaida.

Šis įskiepis padeda apsaugoti vieną jautriausių vietų – banko sąskaitą, tačiau jis neužkerta kelio kibernetiniams nusikaltėliams, kurie bando pasipelnyti kitose srityse.

4.4. Cloudphish Anti-Phishing Extension

Fišing puolimų per elektroninį paštą apsisaugojimo priemonė. Kiekvienas įskiepio vartotojas yra identifikuojamas ir gali nusistatyti patikimus adresus ir domenus. Visi ateinantys laiškai ateinantys ne iš vartotojui patikimų šaltinių yra įvertinami ir jų tinkamumas parodomas vartotojui.

„Cloudphish“ padeda apsaugoti nuo išpuolių per vieną populiariausių vietų – naršyklėje naudojamą elektroninį paštą, tačiau į kenkėjiškas svetaines galima patekti ir paprasčiausiai naršant internete ar jeigu naudojamos programos valdyti pranešimus, pavyzdžiui „Microsoft Outlook“ ar pašto dėžutę integruota operacinėje sistemoje.

lentelė 1 Pagrindiniai rašto darbo stiliai ir jų aprašymai (Rašto darbo šablonas, KTU, 2019)

Stiliaus pavadinimas	Stiliaus pavadinimas galerijoje	Stiliaus formalieji reikalavimai	Stiliaus naudojimo aprašymas
Antraštė non-TOC	Antraštė non-TOC	Šrifto dydis 12 pt, šriftas paryškintas, intervalas tarp eilučių – 1,15, atstumas prieš ir po antraštės – 10 pt, centruota lygiuotė.	Antraštėms, kurios nėra įtraukiamos į turinį: „Turinys“.
Antraštė be nr.	Antraštė be nr.	Šrifto dydis 12 pt, šriftas paryškintas, intervalas tarp eilučių – 1,15, atstumas prieš ir po antraštės – 10 pt, centruota lygiuotė, antraštė rašoma naujame puslapyje – po puslapio skirtuko.	Antraštėms, kurios įtraukiamos į turinį, bet nėra numeruojamos: „Lentelių sąrašas“, „Paveikslų sąrašas“, „Santrumpų ir terminų sąrašas“, „Įvadas“, „Išvados“, „Literatūros sąrašas“, „Informacijos šaltinių sąrašas“, „Priedai“.
1. Heading 1,Skyrius	Skyrius	Šrifto dydis 12 pt, šriftas paryškintas, intervalas tarp eilučių – 1,15, atstumas po antraštės – 10 pt, abipusė lygiuotė, antraštė rašoma naujame puslapyje – po puslapio skirtuko.	Skyrių antraštėms, kurios įtraukiamos į turinį ir yra numeruojamos.
1.1. Heading 2,Poskyris	Poskyris	Šrifto dydis 12 pt, šriftas paryškintas, intervalas tarp eilučių – 1,15, atstumas prieš ir po antraštės – 10 pt, abipusė lygiuotė, numeracija siejama su aukštesnio lygio antrašte.	Poskyrių antraštėms, kurios įtraukiamos į turinį ir yra numeruojamos.
1.1.1. Heading 4,Skyrelis	Skyrelis		Skyrelių antraštėms, kurios įtraukiamos į turinį ir yra numeruojamos.
Tekstas	Tekstas	Šrifto dydis 12 pt, intervalas tarp eilučių – 1,15, atstumas po pastraipos – 10 pt, abipusė lygiuotė.	Tekstui visose rašto darbo dalyse (įvade, skyriuose, poskyriuose ir t.t.).
List Bullet;Sąrašas (suženklintas)	Sąrašas (suženklintas)	Pirmos pastraipos eilutės įtrauka – 0,63 cm, šrifto dydis 12 pt, intervalas tarp eilučių – 1,15, atstumas tarp tokio paties stiliaus pastraipų – 0 pt, atstumas po sąrašo – 10 pt, abipusė lygiuotė.	Tekstui, kuris pateikiamas suženklintu sąrašu.
List Number;Sąrašas (numeruotas)	Sąrašas (numeruotas)	Šrifto dydis 12 pt, intervalas tarp eilučių – 1,15, atstumas tarp tokio paties stiliaus pastraipų – 0 pt, atstumas po sąrašo – 10 pt, abipusė lygiuotė.	Tekstui, kuris pateikiamas sunumeruotu sąrašu.
Footnote Text;Išnašos tekstas	Išnašos tekstas	Šrifto dydis 10 pt, intervalas tarp eilučių – 1,15, atstumas prieš ir po sąrašo – 0 pt, abipusė lygiuotė.	Tekstui, kuris pateikiamas išnašose.
Lentelės pavad.	Lentelės pavad.	Šrifto dydis 11 pt, intervalas tarp eilučių – 1,15, atstumas prieš pavadinimą – 10 pt, po	Lentelių pavadinimams (numeris ir žodis lentelė rašomas paryškintu šriftu).

Stiliaus pavadinimas	Stiliaus pavadinimas galerijoje	Stiliaus formalieji reikalavimai	Stiliaus naudojimo aprašymas
		pavadinimo – 3 pt, lygiuotė prie kairiojo krašto.	
Lentelės I eil.	Lentelės I eil.	Šrifto dydis 10 pt, šriftas paryškintas, intervalas tarp eilučių – 1,15, atstumas prieš ir po pastraipos – 3 pt, lygiuotė prie kairiojo krašto.	Tekstui lentelės antraštinei (pirmai) eilutei.
Lentelė	Lentelė	Šrifto dydis 10 pt, intervalas tarp eilučių – 1, atstumas prieš ir po pastraipos – 3 pt, lygiuotė prie kairiojo krašto.	Tekstui lentelėje.
Caption,Paveikslo pavad.	Paveikslo pavad.	Šrifto dydis 11 pt, intervalas tarp eilučių – 1,15, atstumas prieš ir po pavadinimo – 10 pt, centruota lygiuotė.	Paveikslų pavadinimams (numeris ir santrumpa pav. rašoma paryškintu šriftu).
Figure;Paveikslas	Paveikslas	Atstumas prieš ir po paveikslo – 10 pt, centruota lygiuotė.	Paveikslui, iliustracijai .
Bibliography,Bibliografija	Bibliografija	Šrifto dydis 12 pt, intervalas tarp eilučių – 1,15, atstumas tarp tokio paties stiliaus pastraipų – 2 pt, abipusė lygiuotė.	Literatūros ir Informacijos šaltinių sąrašuose nurodytiems šaltiniams.
Priedas	Priedas	Šrifto dydis 12 pt, šriftas paryškintas, intervalas tarp eilučių – 1,15, atstumas prieš ir po antraštės – 10 pt, lygiuotė prie kairiojo krašto.	Priedo numeriui, žodžiui *priedas*, priedo pavadinimui.
TOC 1,Turinys 1	Turinys 1	Šrifto dydis 12 pt, šriftas paryškintas, intervalas tarp eilučių – 1,15, įtrauka – 0,64.	Turinyje esančioms antraštėms, kurios nėra numeruojamos („Lentelių sąrašas“, „Paveikslų sąrašas“, „Santrumpų ir terminų sąrašas“, „Įvadas“, „Išvados“, „Literatūros sąrašas“, „Informacijos šaltinių sąrašas“, „Priedai“) ir numeruojamai antraštei „Skyriaus pavadinimas“.
TOC 2,Turinys 2	Turinys 2	Šrifto dydis 12 pt, intervalas tarp eilučių – 1,15, įtrauka – 0,96.	Turinyje esančiai antraštei „Poskyrio pavadinimas“.
TOC 3,Turinys 3	Turinys 3	Šrifto dydis 12 pt, intervalas tarp eilučių – 1,15, įtrauka – 1,28.	Turinyje esančiai antraštei „Skyrelio pavadinimas“.
TOC 4,Turinys 4	Turinys 4	Šrifto dydis 12 pt, intervalas tarp eilučių – 1,15, įtrauka – 0,64.	Numeruotiems priedams.

Išvados (pavyzdys)

1. Pagal paskutinių metų fišingo puolimų atskaita matome, kad ši problema yra aktuali ir jis dažnėja.
2. Visas kenkėjiškas svetaines įmanoma atpažinti pagal jų URL adresą ir yra daug skirtingų būdų, kaip galima tai padaryti, tačiau nei vienas būdas nėra 100% tikslus.
3. Populiariausias ir geriausius rezultatus demonstruojantis algoritmas yra „Random Forest“.
4. Rinkoje jau yra programų padedančių apsisaugoti nuo fišingo puolimų, tačiau jos yra arba pritaikytos specifiniams atvejams arba turi trūkumų, kuriuos ši programa gali išspręsti.

Literatūros sąrašas (pavyzdys)

[1] Adewole, K. S., Akintola, A. G., Salihu, S. A., Faruk, N., & Jimoh, R. G. (2019). Hybrid Rule-Based Model for Phishing URLs Detection. Emerging Technologies in Computing, 119–135. doi:10.1007/978-3-030-23943-5_9

[2] Althobaiti, K., Rummani, G., & Vaniea, K. (2019). A review of human-and computer-facing url phishing features. Paper presented at the Proceedings - 4th IEEE European Symposium on Security and Privacy Workshops, EUROS and PW 2019, 182-191. doi:10.1109/EuroSPW.2019.00027

[3] Anand, A., Gorde, K., Antony Moniz, J. R., Park, N., Chakraborty, T., & Chu, B. -. (2019). Phishing URL detection with oversampling based on text generative adversarial networks. Paper presented at the Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018, 1168-1177. doi:10.1109/BigData.2018.8622547

[4] Aydin, M., & Baykal, N. (2015). Feature extraction and classification phishing websites based on URL. Paper presented at the 2015 IEEE Conference on Communications and NetworkSecurity, CNS 2015, 769-770. doi:10.1109/CNS.2015.7346927

[5] APWG, “Phishing Activity Trends Report, 9nd Quarter 2019,” Tech. Rep. November 4, 2019.

[6] APWG, “Phishing Activity Trends Report, 9nd Quarter 2019,” Tech. Rep. November 4, 2019.

[7] Bahnsen, A. C., Bohorquez, E. C., Villegas, S., Vargas, J., & Gonzalez, F. A. (2017). Classifying phishing URLs using recurrent neural networks. Paper presented at the ECrime Researchers Summit, eCrime, 1-8. doi:10.1109/ECRIME.2017.7945048

[8] Basnet, R. B., & Doleck, T. (2015). Towards developing a tool to detect phishing URLs: A machine learning approach. Paper presented at the Proceedings - 2015 IEEE International Conference on Computational Intelligence and Communication Technology, CICT 2015, 220-223. doi:10.1109/CICT.2015.63

[9] Buber, E., Diri, B., & Sahingoz, O. K. (2017). Detecting phishing attacks from URL by using NLP techniques. Paper presented at the 2nd International Conference on Computer Science and Engineering, UBMK 2017, 337-342. doi:10.1109/UBMK.2017.8093406

[10] Buber, E., Diri, B., & Sahingoz, O. K. (2018). NLP based phishing attack detection from URLs doi:10.1007/978-3-319-76348-4_59

[11] Daeef, A. Y., Ahmad, R. B., Yacob, Y., & Phing, N. Y. (2017). Wide scope and fast websites phishing detection using URLs lexical features. Paper presented at the 2016 3rd International Conference on Electronic Design, ICED 2016, 410-415. doi:10.1109/ICED.2016.7804679

[12] Feroz, M. N., & Mengel, S. (2015). Phishing URL detection using URL ranking. Paper presented at the Proceedings - 2015 IEEE International Congress on Big Data, BigData Congress 2015, 635-638. doi:10.1109/BigDataCongress.2015.97

[13] Gupta, S., & Singhal, A. (2018). Dynamic classification mining techniques for predicting phishing URL doi:10.1007/978-981-10-5699-4_50

[14] Gupta, S., & Singhal, A. (2018). Phishing URL detection by using artificial neural network with PSO. Paper presented at the 2nd International Conference on Telecommunication and Networks, TEL-NET 2017, , 2018-January 1-6. doi:10.1109/TEL-NET.2017.8343553

[15] Hawanna, V. R., Kulkarni, V. Y., & Rane, R. A. (2017). A novel algorithm to detect phishing URLs. Paper presented at the International Conference on Automatic Control and

Dynamic Optimization Techniques, ICACDOT 2016, 548-552. doi:10.1109/ICACDOT.2016.7877645

[16] Jain, A. K., & Gupta, B. B. (2018). PHISH-SAFE: URL features-based phishing detection system using machine learning doi:10.1007/978-981-10-8536-9_44

[17] Jeeva, S. C., & Rajsingh, E. B. (2016). Intelligent phishing url detection using association rule mining. Human-Centric Computing and Information Sciences, 6(1) doi:10.1186/s13673-016-0064-3

[18] Li, Y., Yang, Z., Chen, X., Yuan, H., & Liu, W. (2019). A stacking model using URL and HTML features for phishing webpage detection. Future Generation Computer Systems, 94, 27-39. doi:10.1016/j.future.2018.11.004

[19] Parekh, S., Parikh, D., Kotak, S., & Sankhe, S. (2018). A new method for detection of phishing websites: URL detection. Paper presented at the Proceedings of the International Conference on Inventive Communication and Computational Technologies, ICICCT 2018, 949-952. doi:10.1109/ICICCT.2018.8473085

[20] Pradeepthi, K. V., & Kannan, A. (2015). Performance study of classification techniques for phishing URL detection. Paper presented at the 6th International Conference on Advanced Computing, ICoAC 2014, 135-139. doi:10.1109/ICoAC.2014.7229761

[21] Rao, R. S., Vaishnavi, T., & Pais, A. R. (2019). CatchPhish: Detection of phishing websites by inspecting URLs. Journal of Ambient Intelligence and Humanized Computing, doi:10.1007/s12652-019-01311-4

[22] Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, 345-357. doi:10.1016/j.eswa.2018.09.029

[23] Shrestha, N., Kharel, R. K., Britt, J., & Hasan, R. (2015). High-performance classification of phishing URLs using a multi-modal approach with MapReduce. Paper presented at the Proceedings - 2015 IEEE World Congress on Services, SERVICES 2015, 206-212.