# МОДИФИЦИРОВАННЫЙ АЛГОРИТМ ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ОБНАРУЖЕНИЯ И КЛАССИФИКАЦИИ ФИШИНГОВЫХ АТАК

**A.Maraximov**
TerDU professori
**K.Xudaybergenov**
O'zbekiston Milliy universiteti dotsenti
**H.Choriyev**
**A.Nasiriddinov**
Termiz davlat universiteti tayanch doktorantlari

***https://doi.org/10.5281/zenodo.7403080***

***Аннотация***. *Фишинговые веб-сайты относятся к атаке, когда киберпреступники подделывают официальные веб-сайты, чтобы заманить людей к доступу, чтобы незаконно получить личность пользователя, пароль, конфиденциальность и даже свойства. Эта атака представляет большую угрозу для не опытных пользователей сети Интернет и становится все более и более сложнее. Многие предложения по выявлению фишинговых веб-сайтов показали свою эффективность и преимущество методов для обнаружения и классификации, унифицированной указателя ресурсов (URL). Хотя для обнаружения и классификации фишинговых атак было предложено несколько подходов, подходы машинного обучения и искусственного интеллекта на основе URL-адресов обеспечивают более высокие результаты производительности, но все они зависят от используемого набора признаков. Для повышения точности фишинга обнаружение веб-сайтов, в статье предлагается новая модель на основе дерево решении и набор признаков для устройств Интернета вещей (IoT), которые имеет ограниченные возможности и низким энергопотреблением. В настоящем исследовании рассматривается, как выбор набора признаков из обучающихся набор данных, который существенно повышает скорость и производительность классификации фишинговых атак в устройствах IoT. Согласно экспериментальным и сравнительным результатам реализованных алгоритмов классификации, алгоритм кусочно линейный дерево решении, основанными на новых функции активации, обеспечивает наилучшую производительность с точностью 97,50% для обнаружения фишинговых URL-адресов.*

***Ключевые слова:*** *дерево решении, классификатор, фишинговая атака, URL-ресурс, обучающая выборка, машинное обучение.*

# A MODIFIED ALGORITHM FOR INCREASING THE PERFORMANCE OF MACHINE LEARNING FOR PHISHING ATTACK DETECTION AND CLASSIFICATION

***Abstract***. *Phishing websites refer to an attack where cyber criminals spoof official websites to lure people into accessing to illegally obtain user identity, password, privacy, and even properties. This attack poses a great threat to inexperienced Internet users and is becoming more and more difficult. Many proposals for detecting phishing websites have shown their effectiveness and the advantage of methods for detecting and classifying Uniform Resource Locators (URLs). Although several approaches have been proposed for detecting and classifying phishing attacks, URL-based machine learning and artificial intelligence approaches provide better performance results, but they all depend on the feature set used. To improve the accuracy*

*of phishing website detection, the article proposes a new decision tree-based model and feature set for Internet of Things (IoT) devices that have limited capabilities and low power consumption. The present study examines how the selection of a feature set from a trainee data set significantly improves the speed and performance of classifying phishing attacks in IoT devices. According to the experimental and comparative results of the implemented classification algorithms, the piecewise linear decision tree algorithm based on the new activation functions provides the best performance with 97.50% accuracy for detecting phishing URLs.*

***Keywords**: decision tree, classifier, phishing attack, URL resource, training set, machine learning.*

**1. Введение**

С появлением сети Интернет стало ясно, что происходит новая технологический прогресс в киберпреступлениях. В рамках этого прогресса многие сферы бизнеса и информационных технологии перешли от традиционных сервисов к онлайновым формам. Наряду с информационной технологией и использованием преимущества повсеместного распространения онлайн операции многие правонарушения также переместились в онлайн формам, то есть киберпреступления. На сегодняшний день одним из самых распространенных вариантов кибератак является атаки фишинга. В 2019 году глобальное мошенничество с онлайн-банковские транзакции составило 1920 млн долларов, из них 318 млн долларов приходится на фишинговые атаки [1], что делает его одним из самых эффективных и распространённых мошенничеств в сети Интернет [2].

Обычных пользователей веб-броузеров просить вводит свои личные данные во время фишинговой атаки, обычно через унифицированный указатель ресурсов (URL). Как правило, URL-адрес в веб-броузере, использующийся в фишинговой атаке, маскируют, используя длинные последовательности буквенно-цифровых символов и/или вводя символы, похожие на исходный URL-адрес (например, www.mibank.com вместо www.mybank.com). Если вредоносный URL-адрес доставляется на устройства с маленькими экранами (например, мобильные телефоны, планшеты, гаджеты и т.д.), фишинговая атака становиться еще более эффективна и результативным для кибермошенников [8]. В веб-броузерах адресная строка для ввода адреса веб-сайта обычно уменьшена или иногда скрыта от пользователя сети Интернет. Такие устройства составляют набор устройств так называемый Интернет вещей (англ. Internet of Things, IoT) [3]. Многие устройства IoT используются для обмена сообщениями, документами, слушание онлайн радио, просмотр фильмов, покупки товаров в сети Интернет, общения с друзьями/коллегами и т.д. Учитывая эти факты, что цели кибератак считается переходит на устройства IoT и их пользователей становиться все больше и больше [11]. Кроме того, фишинговая кибератака, которая, как ожидается, будет расти быстрее, чем любые другие, который очень привлекателен для киберпреступников из-за физических особенностей и низкой уровня безопасности таких устройств как IoT.

Учитывая особенности фишинговых атак, исследование сосредоточено на изучении признаков в обучающих наборов данных для повышения производительности алгоритмов машинного обучения и искусственного интеллекта для обнаружения и классификации фишинга. Однако, в многих работах по обнаружению и классификации фишинговых атак, большинство из них были сосредоточены на определении того, какой

классификатор работает лучше с учетом предварительно определенных признаков, полученных с помощью сторонних сервисов и источников обучающих набор данных, которые находится в общедоступных репозиториях [4]. В этих работах также используются сложные структуры данных и представления данных в сочетании с интенсивными вычислительными процессами, что делает их непригодными для использования в устройствах IoT. Кроме того, некоторые работы приобретают признаки посещения подозрительной веб-страницы, подразумевая, что они стали жертвой атаки. Устройства IoT характеризуются ограниченными вычислительными возможностями и низким энергопотреблением, что делает не пригодным таких методов и алгоритмов классификации для использования в этих системах [5].

В таких случаях такие алгоритмы классификации, работающие в устройствах IoT, должны быть легкими, энергоэффективными и рекомендуется избегать использования сложных структур данных, а используемые источники набор обучающих данных и их признаки должны быть настолько простыми, насколько это возможно. Учитывая вышеуказанные требования, в этой работе предлагается метод на основе дерево решении который предложено в [6] для обнаружения фишинговых URL-адресов в средах IoT, которое максимально увеличивает скорость обнаружения и точность классификации фишинговых атак. Выбор набора признаков имеет решающее значение для предложения подхода к обнаружению фишинга, применимого на практике. Кроме того, предлагаемый метод позволяет обнаруживать атаки в режиме реального времени и атаки нулевого дня, не зависит от сторонних сервисов.

Основными вкладами этой статьи являются:

1. Выбор признаков из набор обучающих данных для обнаружения и классификации фишинговых URL-адресов, именно подходящее для систем IoT;

2. Служить отправной точкой для исследователей и практиков в разработке решений задач классификации фишинговых атак для систем с ограниченными свойствами как IoT.

## 2. Основные понятия и методы для фишинговых атак

### *2.1. Фишинговые атаки*

Фишинговые атаки могут осуществляться через URL-адреса из веб-броузера пользователей. Как правило, фишинговые атаки на основе URL-адресов в основном выполняются путем встраивания специальных слов и/или символов в URL-адреса:

(а) генерируется похожие слова, но с не значительными ошибками;

(б) содержать набор специальных символов/букв для перенаправления веб-страницы;

(в) применяют укороченные и/или излишне слишком длинные URL-адреса, не пригодным для понимая;

(д) используют привлекательные ключевые слова, которые кажутся правильными;

(е) в большинство случаях добавляются в ссылку вредоносный файл, который после автоматической скачивания переходит в устройство IoT жертва-пользователя.

Одним и подходов к обнаружению и классификации фишинговых URL-адресов основан на черных списках, которые опираются на репозиторий уже классифицированных веб-сайтов (*https://phishtank.com*). Этот подход является высокоскоростным и эффективным, но имеет некоторые недостатки. Например, URL-адрес, не существующий

в наборе обучающих данных, не будет правильно классифицирован, особенно URL-адреса атак нулевого дня. В таких подходах, обнаруженных в методах на основе черных списков, что традиционные алгоритмы машинного обучения достаточно хорошо решает задачи с проблемами обнаружения фишинговых URL-адресов [7-11]. Эти требования особенно подходят для применения для устройств IoT из-за их относительно низкой вычислительной мощности и ресурсов.

*2.2. URL-адрес*

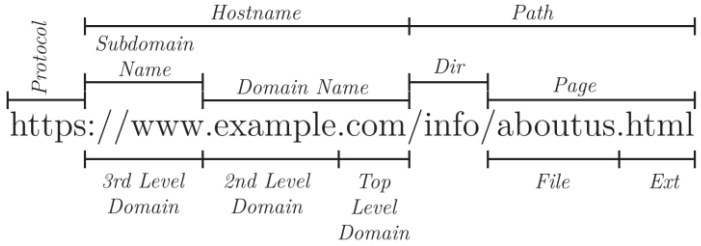
Данное исследование связано с фишинговые атаками, поэтому в настоящей работе анализируется данные относящейся адресам, который называется унифицированным указателем ресурсов (URL), который можно найти в стандарте RFC1738. Общий вид URL-адресов показаны на Рис. 1.

*2.3. Методы и алгоритмы для классификации фишинговых URL-адресов*

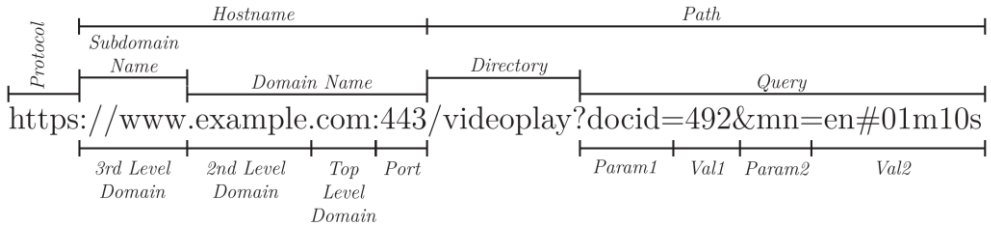
Широко распространенная методика, используемая для фишинговых атак это создание очень большого количество URL-адресов с разными всевозможными вариантами. Это дает возможность отвлекает внимание обычных Интернет-пользователей, и это значит, что вероятность успешной фишинговой атаки возрастает в несколько раз порядок. В URL-адресной строке вводится несколько косых черт, указывающих на несколько каталогов в URL-адресе и выглядящих правильными для не опытных пользователей IoT устройств. Точно так же введение нескольких точек и некоторых символов/цифр в доменное имя для создания нескольких субдоменов создает ощущение правильного URL-адреса.

Такие сгенерированные вредоносные URL-адреса очень часто заменяют буквенно-цифровые символы другими, т.е. символами Unicode и/или шестнадцатеричным представлением символов. Текст английского языка имеет относительно низкую энтропию, т.е. он предсказуем, и при введении разных символов энтропия меняется сильнее. Следовательно, использование энтропии может привести к обнаружению и правильной классификации вредоносных URL-адресов.

Учитывая такие аспекты фишинговых атак, в этой статье основное внимание уделяется предложению облегченного представления URL-адресов и более производительного алгоритма для обнаружения и классификации фишинговых URL-ресурсов в системах IoT. Киберпреступники во время фишинговой атаки очень часто доставляет вредоносный URL-адрес с помощью обычных приложении (электронной почты, Telegram, Tweeter, Facebook и т.д.). Если не опытный пользователь сети Интернета получает доступ к фишинговому URL-адресу, вредоносные действия будет работать на пользу киберпреступника.




(a) URL directing to a file named aboutus.html




(b) URL with queries.

*Рис. 1. Общий вид формы унифицированного указателя ресурсов и его частей*

**3. Выбор признаков для обнаружения фишинговых атак**

В работах [12-17] авторы исследовали некоторые подходы обнаружения фишинговых URL-адресов. Эти подходы используют несколько признаков, полученных из URL-адресов на практике. В этом исследовании предлагаемый набор признаков был построен с учетом характера фишинговых атак и проецирования их на URL-адреса, например, фишеры пытаются запутать обычных пользователей устройств IoT, делая URL-адреса не пригодным для чтения и понимания. В итоге, сгенерированные фишинговые URL-адреса становится больше по количеству и используют другие символы/цифры, чем правильные адреса.

Исходя из этого факта, в этой исследовании рекомендуется использовать такие типи признаков, измеряющих длину некоторых частей URL, количество символов/цифр и признаков связанные с HTTP/S. Для лучшего понимания структуры рекомендуется посмотреть Рис. 1. В этой исследовании рекомендуется использовать следующие набор признаков:

П-1. Длина URL-адреса;
П-2. Длина субдомена;
П-3. Длина домена;
П-4. Длина домена верхнего уровня (TLD);
П-5. Длина имени хоста;
П-6. Длина самого длинного токена в URL-адресе;
П-7. Длина самого короткого токена в URL-адресе;
П-8. Средняя длина маркера;
П-9. Количество цифр в URL;
П-10. Количество токенов в субдомене;
П-11. Количество токенов в домене;
П-12. Количество токенов в домене верхнего уровня (TLD);
П-13. Количество специальных символов в имени хоста;
П-14. Количество косых черт в URL-адресе;
П-15. Количество символов Unicode в URL-адресе;
П-16. Количество точек в URL;
П-17. Количество дефисов в имени хоста;
П-18. Количество параметров в запросе;
П-19. Количество подкаталогов в пути;
П-20. Количество цифр в имени хоста;
П-21. Количество букв в имени хоста;
П-22. Количество символов в имени хоста;

П-23. Указывает, содержит ли URL-адрес IP-адрес;
П-24. Указывает, является ли URL-адрес HTTP или HTTPS;
П-25. Указывает, содержит ли URL-адрес исполняемые файлы;
П-26. Номер используемого порта (если имеется в URL-адресе).

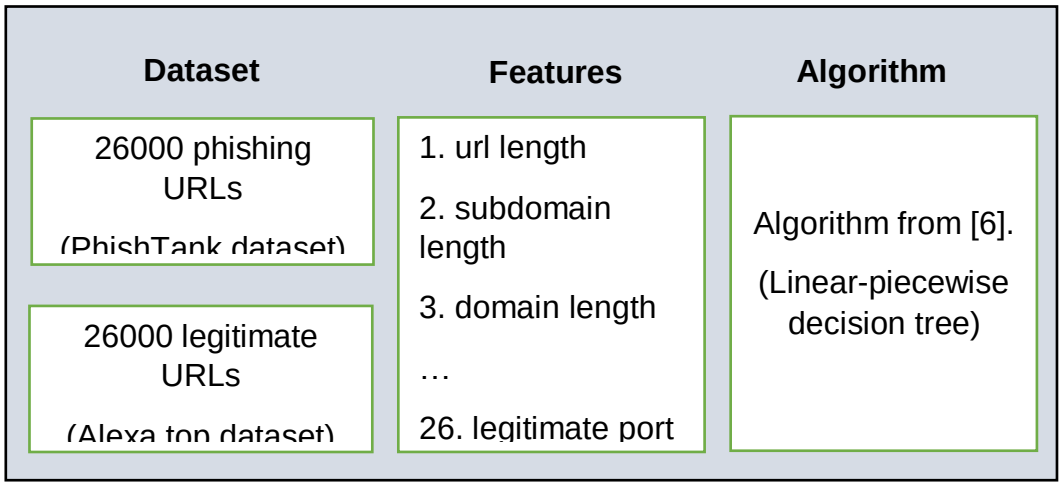



*Рис. 2. Методологическая схема для классификации фишинговых атак*

**4. Вычислительные эксперименты**

В этой статье использовалось восемь различных алгоритмов классификации (Наивный Байес, Случайный лес, kNN, Adaboost, Логистическая регрессия, Нейронные сети, Глубокое обучение и Дерево решений) в качестве механизма машинного обучения предлагаемой системы, а затем проведено сравнительный анализ.

Одной из важных проблем при тестировании предлагаемой системы является использование общепринятого набора данных. Было проведено тест на наборе данных, который содержит 73575 URL-адресов. Этот набор данных содержит 36400 законных URL-адресов и 37175 фишинговых URL-адресов.

Эксперименты проводились на устройстве DELL Alienware с процессором Intel Core i9-12900HK с частотой 3,9 ГГц и 32 ГБ оперативной памяти DDR5. Во время тестов использовалась 10-кратная перекрестная проверка и значения параметров по умолчанию для всех алгоритмов.

Каждый набор тестов выполняется с использованием восьми различных алгоритмов машинного обучения. Матрица неточности для протестированных алгоритмов обучения строится, как показано в Таблице 1.

*Таблица 1.*

*Матрица неточности (П – позитив, Н – негатив)*

| Матрица неточности | | Прогнозированный | |
|---|---|---|---|
| | | П | Н |
| Дерево решении | П | 36728 | 447 |
| | Н | 1252 | 31052 |
| Adaboost | П | 35813 | 1362 |
| | Н | 3609 | 32791 |
| Логистическая регрессия | П | 35652 | 1394 |
| | Н | 3804 | 32721 |
| kNN (k=3) | П | 36214 | 961 |

| | | | |
|---|---|---|---|
| | Н | 2082 | 34318 |
| Случайный лес | П | 36806 | 369 |
| | Н | 1120 | 35280 |
| Наивный Байес | П | 27663 | 9512 |
| | Н | 1247 | 35153 |
| Нейронный сеть | П | 36628 | 347 |
| | Н | 1352 | 32052 |
| Глубокое обучение | П | 36500 | 547 |
| | Н | 1120 | 30012 |

Используя значения в матрице неточности, рассчитываются 4 различные статистики, такие как точность, чувствительность, F-мера и точность, для измерения полезности и эффективности алгоритмов. Эти статистические данные, формулировка которых представлена в уравнениях (1-4), также важны для сравнения проверенных подходов к машинному обучению.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (2)$$

$$F - мера = 2 \times \frac{Precision \times Sensitivity}{Precision \times Sensitivity} \quad (3)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (4)$$

где TP означает истинно положительный результат, TN означает истинно отрицательный результат, FP означает ложноположительный результат, а FN означает ложноотрицательный уровень алгоритмов классификации. В соответствии с этими уравнениями результаты расчетов реализованных алгоритмов машинного обучения представлены в таблице 2 в сравнительном формате.

*Таблица 2.*

*Результаты классификации алгоритмов*

| Алгоритм | Точность (Precision) | Чувствительность (Sensitivity) | F-мера | Точность (Accuracy) |
|---|---|---|---|---|
| Дерево решении | 0.972 | 0.978 | 0.973 | 97.50% |
| Adaboost | 0.908 | 0.963 | 0.935 | 95.16% |
| Логистическая регрессия | 0.936 | 0.936 | 0.936 | 95.27% |
| kNN (k=3) | 0.940 | 0.977 | 0.958 | 95.67% |
| Случайный лес | 0.971 | 0.990 | 0.980 | 92.98% |
| Наивный Байес | 0.928 | 0.975 | 0.951 | 94.25 |
| Нейронный сеть | 0.938 | 0.915 | 0.945 | 96.10 |
| Глубокое обучение | 0.928 | 0.922 | 0.922 | 96.45 |

Как видно из этой таблицы, алгоритм кусочно-линейный дерево решений (КЛ-ДР) [6], имеет наилучшую производительность классификации (точность 97,50%). Кроме того, из этой таблицы также можно увидеть влияние функций.

Этот показатель точности интерпретируется как приемлемый и хороший результат для обнаружения фишинга. 100% точность невозможна. Потому что в то время, как менеджеры по безопасности систем пытаются использовать некоторые новые методы, злоумышленники пытаются улучшить свои методы атаки, чтобы обойти существующие/разрабатываемые антифишинговые системы. В то же время предлагаемый подход зависит от URL-адреса фишинговой веб-страницы. После того, как было изучено необнаруженные фишинговые веб-страницы, получено, что некоторые из этих страниц имеют короткий домен и субдомены без каких-либо путей. Если URL-адрес содержит только одно доменное имя, такое как «www.testname.org», из-за особенностей предлагаемого решения, основанных на NLP, эти страницы в большинстве случаев не могут быть обнаружены. При стандартной фишинговой атаке веб-страница оформлена как легальная веб-страница, поэтому злоумышленники пытались скрыться с использованием длинного URL-адреса, используя специальные слова, чтобы обмануть пользователей. Потому что более короткие URL-адреса могут быть перехвачены пользователями, у которых есть начальные знания о фишинговых атаках.

Чтобы увидеть общее улучшение производительности в сравнительном виде, составлена Таблица 3. Почти во всех алгоритмах машинного обучения, алгоритм КЛ-ДР обеспечивают лучшую производительность для классификации URL-адресов, со средним показателем 10,86%. Кроме того, использование гибридных функций также увеличивает производительность системы примерно на 2,24% в соответствии с функциями NLP и на 13,14% в соответствии с Word Vectors.

*Таблица 3.*

*Сравнительный анализ алгоритмов*

| Алгоритм | Точность |
|---|---|
| Дерево решении | 98.25 |
| Adaboost | 92.20 |
| Логистическая регрессия | 93.50 |
| kNN (k=5) | 81.10 |
| Случайный лес | 92.30 |
| Наивный Байес | 92.43 |
| Нейронный сеть | 95.23 |
| Глубокое обучение | 96.75 |

**5. Заключение**

В этой статье предложено интеллектуальная система обнаружения фишинга, используя алгоритм дерево решений для устройств IoT. Метод предлагаются как упрошенный тип именно для устройств с ограниченными возможностями, который можно гибко применять в различных методах обучения алгоритмов дерево решении. Численные эксперименты подтверждают эффективность предложенного метода для IoT систем, а также демонстрируют его потенциал для повышения производительности даже с более краткими структурами модели по сравнению с существующими методами. Для повышения точности системы обнаружения создание эффективного списка признаков является важной задачей.

**REFERENCES**

1. R.J. Anderson, C. Barton, R. Böhme, R. Clayton, M.J.G. van Eeten, M. Levi, T. Moore, S. Savage, in: R. Böhme (Ed.), Measuring the Cost of Cybercrime, The Economics of Information Security and Privacy, Springer, 2013, pp. 265-300.
2. Q. Cui, G.-V. Jourdan, G.V. Bochmann, I.-V. Onut, J. Flood, Phishing Attacks Modifications and Evolutions, in: J. Lopez, J. Zhou, M. Soriano (Eds.), Computer Security, Springer International Publishing, Cham, 2018, pp. 243-262.
3. L. Atzori, A. Iera, G. Morabito, Understanding the Internet of Things: definition, potentials, and societal role of a fast evolving paradigm, Ad Hoc Networks. 56 (2017) 122-140.
4. O.K. Sahingoz, E. Buber, O. Demir, B. Diri, Machine learning based phishing detection from URLs, Expert Systems and Applications. 117 (2019) 345-357.
5. M.T. Suleman, S.M. Awan, Optimization of URL-Based Phishing Websites Detection through Genetic Algorithms, Automatic Control and Computer Sciences. 53 (2019) 333-341.
6. Marakhimov A.R., Kudaybergenov J.K., Khudaybergenov K.K., Ohundadaev, U.R. A multivariate binary decision tree classifier based on shallow neural network. Scientific and Technical Journal of Information Technologies, Mechanics and Optics, 2022, 22(4), pp. 725-733.
7. S. Afroz, R. Greenstadt, Phishzoo: Detecting phishing websites by looking at them, in: Fifth IEEE International Conference on Semantic Computing, 2011.
8. S. Sheng, B. Wardman, G. Warner, L. Cranor, J. Hong, An empirical analysis of phishing blacklists, 2009.
9. S. Haruta, H. Asahina, I. Sasase, Visual similarity-based phishing detection using image and CSS with target website finder, in: IEEE Global Communications Conference, 2017.
10. S. Abdelnabi, K. Krombhoiz, M. Fritz, WhiteNet: Phishing website detection by visual whitelists, in: Cryptography and Security, 2019.
11. Peng, I. Harris, Y. Sawa, Detecting phishing attacks using natural language processing and machine learning, in: IEEE 12th International Conference on Semantic Computing(ICSC), 2018.
12. Yazan Ahmad Alsariera, et al., Ai meta-learners and extra-trees algorithm for the detection of phishing websites, IEEE Access 8 (2020) 142532-142542.
13. A. Tewari, B.B. Gupta, Security, privacy and trust of different layers in Internet-of-Things (IoTs) framework, Future Gener. Comput. (2020).
14. Sahingoz OK, Buber E, Demir O, Diri B. Machine learning based phishing detection from URLs. Expert Systems and Applications. 2019. Vol.117, pp.345-57.
15. Shekokar NM, Shah C, Mahajan M, Rachh S. An ideal approach for detection and prevention of phishing attacks. Procedia Computer Science. 2015. pp.49-82.
16. Nezhad JH, Jahan MV, Tayarani-N M-H, Sadrnezhad Z. Analyzing new features of infected web content in detection of malicious webpages. ISC International Journal of Information Security. 2017. Vol. 9(2), pp. 63-83.
17. Rao RS, Pais AR. Detection of phishing websites using an efficient feature-based machine learning framework. Neural Computing and Applications. 2019. vol. 31, pp.3851-3873.