УК БАН '20

Рансъмуер и Устойчиви Стратегии за Архивиране

Автор: Никола Стайков

Ментор: Явор Папазов

6 декември 2019 г.

Съдържание

Абстракт

Рансъмуер е вид компютърен вирус, който критптира файловете на дадена система и изисква да бъде платен откуп, за да бъдат декриптирани. Създателите на рансъмуер могат да правят малки проучвания преди да започнат основната кампания с цел да определят гореспоменатото разпределение. Първата част на този проект разглежда модел, чрез който да бъдат определени оптималните параметри за едно такова проучване. Главната и най-ефективна защита срещу рансъмуер е правенето на архиви. Те на свой ред обаче могат да представляват съществен разход за големите компании, поради което трябва да бъдат внимателно планирани. Това е взето предвид във втората част на проекта, в която е разгледан модел за архивиране на данни, състоящ се от пълни и инкрементални архиви, и е изчислена очакваната цена за възстановяване на данните. Процесът по възстановяване е пресъздаден и анализиран чрез визуализация на python и Монте Карло симулация.

1 Въведение

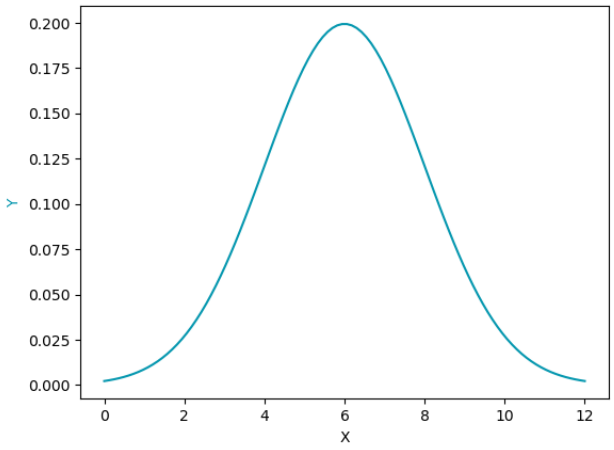
Този проект е разделен на две основни части, разглеждащи съответно модели за оптимизиране на откуп и оптимизиране на архивиране. Изследвайки обстойно двете противоположни позиции, можем да създадем пълна представа за стратегиите едновременно на атакуващите и жертвите. Обмисляйки начини как всяка от двете страни може да подобри стратегията си ни дава идея как да направим стабилна защитна стратегия чрез архивиране.

2 Теория

В тази част са включени всички дефиниции и концепции, които са нужни за цялостното разбиране на проекта.

Дефиниция 1. *Нормално разпределение*, означено с $N(\mu, \sigma)$, е вид непрекъснато разпределение, където с μ, σ и σ^2 са означени средното аритметично, стандартната девиация и вариацията съответно.

Графиката на тази функция образува крива, често наричана също камбанна крива. Тя има максимум $(x, f(x))$ в $\left(\mu, \frac{1}{\sigma\sqrt{2\pi}}\right)$:



Дефиниция 2. Разглеждаме нормално разпределение $N(\mu, \sigma)$. *Стандартната стойност*, или *Z-score*, на дадено x показва колко стандартни девиации е то от дадената средна стойност. Пресмята се по формулата $\frac{x-\mu}{\sigma}$.

Дефиниция 3. За дадено разпределение *функцията на разпределение* $F(x)$ показва вероятността стохастична променлива, следваща разпределението, да е по-малка или равна на x

$$F_X(x) = \mathbb{P}(x \leq X).$$

Дефиниция 4. *Плътност на разпределение* на непрекъсната стохастична променлива x, описва вероятността дадена стохастична променлива x да се окаже в произволен интервал. Формално се дефинира чрез

$$\mathbb{P}(x < X \leq x + \Delta) = F_X(x + \Delta) - F_X(x)$$
$$f_X(x) = \lim_{\Delta \to 0} \frac{F_X(x + \Delta) - F_X(x)}{\Delta}.$$

Дефиниция 5. *Грешка от първи род* е резултат на интегрирането на нормално разпределение, тя приема z-score като параметър и пресмята интеграла между фиксирана точка и средната стойност за разпределението.

$$\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt.$$

Дефиниция 6. *Опит на Бернули* е стохастичен експеримент с два изхода и фиксирани вероятности за провал и успех:

$$P(\text{успех}) = p$$
$$P(\text{провал}) = 1 - p.$$

Дефиниция 7. *Биномно разпределение* е статистическото разпределение на изходите (успех/провал) при провеждането на определен брой опити на Бернули. За n опита и вероятност за успех p вероятността точно k от тях да са успешни е:

$$P(\text{успех} = k) = \frac{\binom{n}{k} p^k (1-p)^{n-k}}{2^k}$$

3 Оптимизиране на откуп

3.1 Въведение

Рансъмуер се появява за първи път през 1989 под формата на the AIDS Troyan, познат също като PC Cyborg. The AIDS Trojan е бил доста лесен за преодоляване, тъй като използва симетрична криптография, и скоро са били разработени начини файловете да бъдат декриптирани, но този случай поставя началото на развитието на много от модерните заплахи. С навлизането на Интернет, рансъмуер се завръща с нова сили, а именно с the Archiveus Trojan и GPcode от 2006. Друг повратен момент в историята на рансъмуер е създаването на биткойн, и крипто-валутите като цяло, по много причини, някои от тях бидейки анонимността и автоматичните и невъзвръщаеми транзакции[1].

В изминалите години е имало опити да бъде направен модел на пазара на malware. В [2], авторите са създали теоретичен модел, взимайки предвид броя потребители, които имат архиви, както и други фактори като разпространението на информация и надеждност на рансъмуер. В [3] е изследван различен подход, който разглежда възможността за допълнително уговаряне на цената като игра между жертвата и престъпниците. Тази разработка се фокусира на теория на игрите и комбинаторика.

Доста усилия са положени и за проследяването на плащания, свързани с рансъмуер в блокчейн, тъй като всички те са публични. В резултат на това има публични данни, свързани с тези плащания, предоставени от [4] и в [5] човек може да се запознае с много заключенияв, подкрепени с данни, отнасящи се не само до рансъмуер, но и до целия черен пазар.

Моделът в настоящата разработка е базиран на описания в [2], но се фокусира върху оптимизирането на параметри, които не са разгледани в споменатата статия.

3.2 Подход

Този модел описва разпространението на рансъмуер вирус. Намира оптималната цена на откуп за рансъмуер атака, която използва единствено botnets, без ключовия компонент на разпространяване на всеки компютър в мрежата. Този вариант на атаката е сравнително евтин за осъществяване, но има ниска ефективност. Третираме декриптирането на файловете на даден компютър като услуга, а откупа като нейната цена, съответно.

Разглеждаме разпределението на Желанието за плащане (ЖЗП) на дадена тестова група. Това е максималната сума, която някой би платил за данните си. Поставяйки се в позицията на престъпниците се опитваме да открием разпределението чрез изследването на тестови групи от хора и как те реагират на дадена цена. Тези тестове обаче ни струват ценно време тъй като осведомеността на хората се показва постоянно. Искаме да разберем колко и колко големи тестове трябва да провеждаме, така че да направим модел на разпределението с приемлива грешка и в същото време без да губим твърде много време.

За даден размер на тестовата група, изчисляваме грешката на дадена група от "потребители"от математически описаната функция на кривата на търсенето, която извличаме от разпределението на ЖЗП. Започвайки с малка група, постепенно увеличаваме размера на тестовата група, изчислявайки и грешката чрез метода на най-малките квадрати на всяка стъпка.

3.3 Модел

Тук математическата страна на модела е разгледана подробно, показвайки как са достигнати резултатите и закюченията.

3.3.1 Размер на тестовата група и грешка

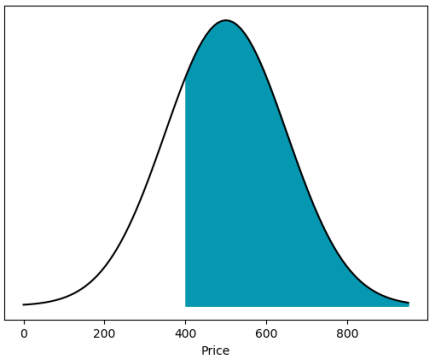
Тази секция описва математическия модел, използвам за оптимизиране на грешката. Изведени са заключения относно размера на тестовата група.

Приемаме, че стойността на данните на хората следва нормална дистрибуция и я свързваме със стохастичната променлива $p \sim N(500, 150)$. Вероятностната плътност (ВП) на нормална дистрибуция $N(\mu, \sigma)$ е

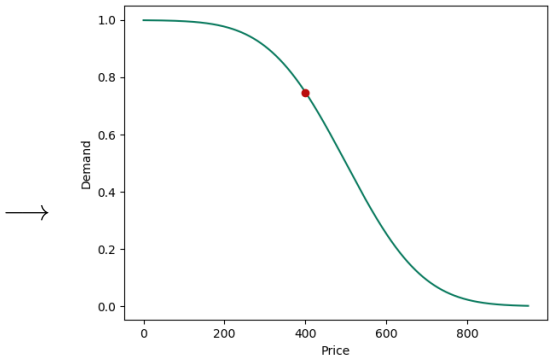
$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

За да изчислим функцията на търсене $f(k)$ от ВП за дадена цена k, трябва да изчислим

$$\int_k^\infty f(x)\,\mathrm{d}\,x.$$



Фигура 1: ВП

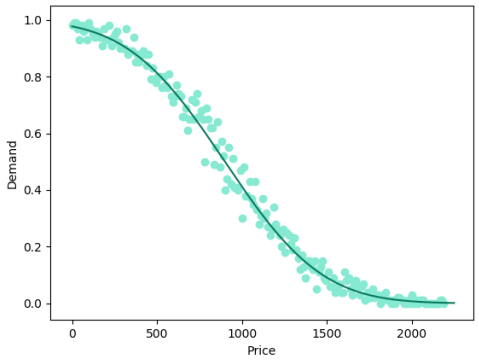


Фигура 2: Цена и търсене

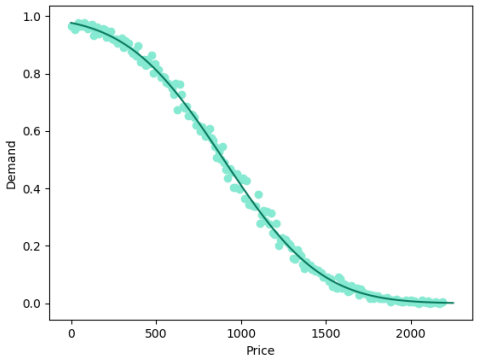
Отбелязаме, че интегралът трябва да бъде изчилсен до безкрайност, но след като k стигне $\mu + 3\sigma$, резултатът става пренебрежимо малък. Правейки това за цялата функция на разпределението получаваме кривата на търсенето чрез процента хора, които биха платили. Нека означим кривата на търсенето с $F(x)$:

$$F(x) = \begin{cases} \frac{1}{2}\left(1 - \operatorname{erf}\left(\frac{z}{\sqrt{2}}\right)\right) \text{ако } x > \mu, \\ \\ \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{z}{\sqrt{2}}\right)\right) \text{ако } x < \mu. \end{cases}$$

Искаме да оптимизираме броя хора във всяка тестова група. Математическата функция, която искаме да опишем ни дава възможността да изчислим грешките от експерименталните данни с максимална точност.

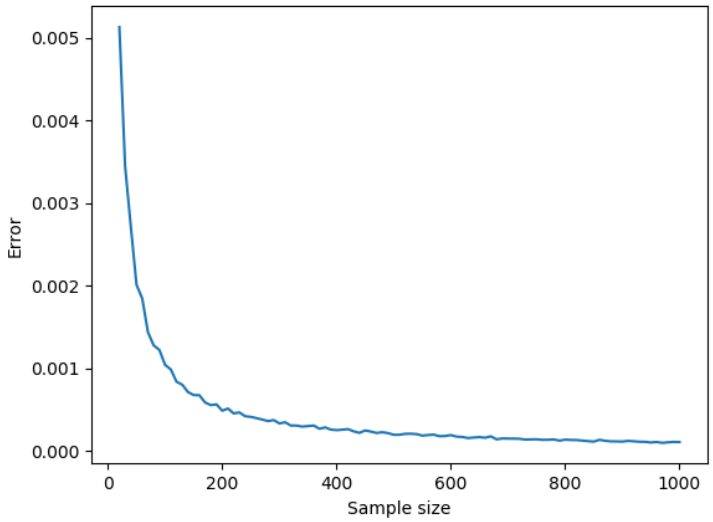


Фигура 3: 100 човека в групата



Фигура 4: 400 човека в групата

Събирайки информация за размера на тестовата груап и грешките, съставяме графика, която показва тези промени.



Фигура 5: Тестов размер и грешка

3.4 Резултати

Изследвахме как размерът на тестовата група влияе на грешката на експерименталните данни и също как процента на защитените с архиви влияе на очакваната печалба. Моделът се фокусира на оптимизирането на цената на откупа, но авторът вярва, че за да можем да предприемем подходящи предпазни мерки срещу атаки от този вид, трябва да разбираме всеки ход на престъпниците. Поставяйки се на мястото на извършителите е ключово за целта. Допълнителни резултати, като връзката между очакваната печалба и броя на хората с архиви може да ни помогне да стигнем до подходящи подходи за справяне със заплахата.

4 Оптимизиране на архивирането

4.1 Въведение

Когато става дума за защита от рансъмуер, най-ефективният метод са архивите. Те на свой ред трябва да бъдат създавани регулярно, защото ефектът от тях при потенциална атака иначе би бил незначителен. Затова е важно протоколите за архивиране да са съставени внимателно и да са съобразени едновременно с рисковете от атака и с нужните за тях ресурси. Нещо повече, оптимизирането на бекъпи може едновременно да увеличи сигурността и да намали разходите.

В тази част на проекта е разгледан модел на архивиране с цел да бъде изчислена очакваната цена. Подобни модели, разглеждащи пълйни и инкрементални архиви, са създавани и изследвани и преди[12][13]. Разглеждаме цикъл от архиви, който се повтаря между два пълни архива и изследваме как интервалите влияят на очакваната цена на възстановяване и колко бързо ефектът, породен от първоначално незащитените данни преди първия пълен архив, намалява с времето.

4.2 Теоретична постановка

Настоящият модел е създаден с идеята да изчисли и оптимизира очакваната цена на възстановяване в случай на атака.

Ще разглеждаме ахивът като структура от данни със следните качества:

$$B \begin{cases} d\text{: датата на архива като разлика в дни от първия архив} \\ p\text{: вероятността възстановяването да е неуспешно} \\ r\text{: цената за опит за възстановяване} \end{cases}$$

Два вида архиви са разгледани:

1. Пълен архив: запазва копие на цялата база данни
2. Инкрементален архив: запазва само промените спрямо последния архив

Атхивите от конкретен вид имат обща вероятност за провал и цена за опит за възстановяване.

За да може един инкрементален архив да е успешен, трябва всички инкрементални архиви преди него до успешен пълен архив също да са успешни, както и самият пълен архив.

В случая цената на данните трябва да се разглежда от субективна гледна точка. Дори и на пазара данните да нямат голяма стойност, ако те са фундаментални за функционирането на компанията, тя ще е готова да плати много за незабавното им възстановяване. Следователно, в описания модел цената на данните се разглежда като вечно увеличаваща се величина и с целите на изследването "скоростта на работа именно цената на данните, генерирани за един ден, на компанията се счита за константа. Ще я означим с w.

Цената на възстановяването на архив ще се разглежда като сума от два фактора:

- Цената на изработването на загубените данни, означена с W
- Цената на процеса по възстановяването, означена с R

. Дефинираме $W = \Delta t.w$, където с Δt означаваме разликата в дни между датана на успешно възстановения архив и датата на атаката, и $R = \sum_{i=1}^{n} r_i$, където броят опити за възстановяване n и

$$S = \Delta t.w + R,$$

Нека разликата в дни между първият архив и датата на атаката е T. В случай, че никой от пълните архиви не се окаже успешен, въвеждаме променлива W_T, съответстваща на цената на преработване на цялата работа от начало. Ясно е, че $W_T > T.w$

4.3 Само пълни архиви

Когато разглеждаме само пълни архиви, моделът е разпределение на Бернули с краен брой опити, а именно броят пълни архиви. Спираме, когато успеем да намерим успешен архив, започвайки от последния и вървейки към първия. Да дефинираме свойствата на пълен архив(B_F):

$$B_F \begin{cases} p_F : \text{вероятността за провал} \\ r_F : \text{цената за опит за възстановяване} \\ t_F : \text{дните между два последователни пълни архива} \end{cases}$$

Нека k е броят пълни архиви направени преди деня на атаката. Тогава:

$$k = \left\lfloor \frac{T}{t_F} \right\rfloor + 1$$

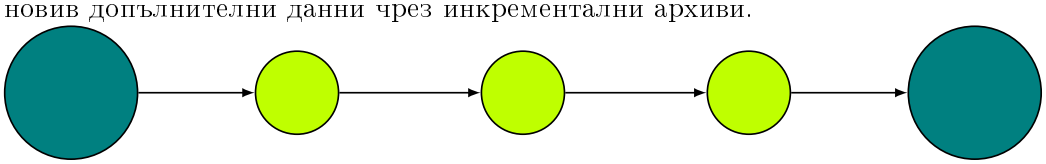
Сега можем да дефинираме очакваната цена на възстановяване:

$$E(T) = p_F^k \left(W_T + k.r_F\right) + \sum_{i=0}^{k-1} (1 - p_F).p_F^i \left(\left(\left\{ \frac{T}{t_F} \right\} + i \right) t_F.w + (i+1).r_F \right) \qquad (1)$$

Направените изчисления са ключови поради невъзможността инкременталните архиви да бъдат възстановени без работещ пълен архив и следователно намирането на такъв е първият ни приоритет. Сега можем да разгледаме инкременталните архиви при работещ пълен архив.

4.4 Инкрементални архиви с работещ пълен архив

Ще разгледаме случая, когато имаме работещ пълен архив и се опитваме да възстановив допълнителни данни чрез инкрементални архиви.



Нека дефинираме свойствата на инкременталните архиви (B_I) по подобен начин:

$$B_I \begin{cases} p_I : \text{вероятността за провал} \\ r_I : \text{цената за опит за възстановяване} \\ t_I : \text{дните между два последователни инкрементални архива} \end{cases}$$

Нека с T_F да означим разликата в дни между денят на атаката и датата на успешния пълен архив и с l да означим броят инкрементални архиви, които трябва да разгледаме. Имаме две опции за l в зависимост от това дали последният пълен архив е бил успешно възстановен:

$$l = \begin{cases} \left\lfloor \frac{T_F}{t_I} \right\rfloor, \text{ако } T_F < t_F \\ \left\lfloor \frac{t_F}{t_I} \right\rfloor - 1, \text{ако } T_F > t_F \end{cases}$$ [1]

Да отбележим, че това последният пълен архив да е успешен е еквивалентно на $T_F < t_F$.

Сега сме в точно обратната ситуация спрямо миналата част. Процесът по възстановяване на инкрементални архиви продължава докато не се натъкнем на провал, тъй като това би означавало, че всички следващи инкрементални архиви също са неизползваеми. С това намаляме W, тъй като в началната позиция сме готови да преработим данните до датата на успешния пълен архив. Сега сме готови да изчислим очакваната цена:

$$f(T_F) = (1-p_I)^l.((T_F - t_I.l).w + r_I.l) + \sum_{i=0}^{l-1}(1-p_I)^i.p_I((T_F - t_I.i)w + r_I.(i+1)) \quad (2)$$

Сега знаем колко ще намалее цената на възстановяването, когато използваме инкрементални архиви, и можем да построим цялостния модел, използвайки уравнения 1 и 2.

4.5 Крайна очаквана цена

Към всяко събираемо в уравнение 1 Трябва да добавим ефектът на инкременталните архиви и получаваме нови събираеми от вида:

$$P(W + R),$$

където P е вероятността определена комбинация от събития да се случи, W е цената на данните, които трябва да бъдат създадени наново, а R е цената на процесът по възстановяването. Инкременталните архиви намаляват цената на данните, които трябва да бъдат създадени наново, но увеличават R. Както споменахме по-горе, има само един случай, вкойто броят инкрементални архиви, които трябва да имаме предвид е различен и той е именно този, в който последният пълен архив е възстановен успешно. Ако i-тият пълен архив е успешен[2]:

$$T_F = t_F\left(\left\{\frac{T}{t_F}\right\} + i - 1\right)$$

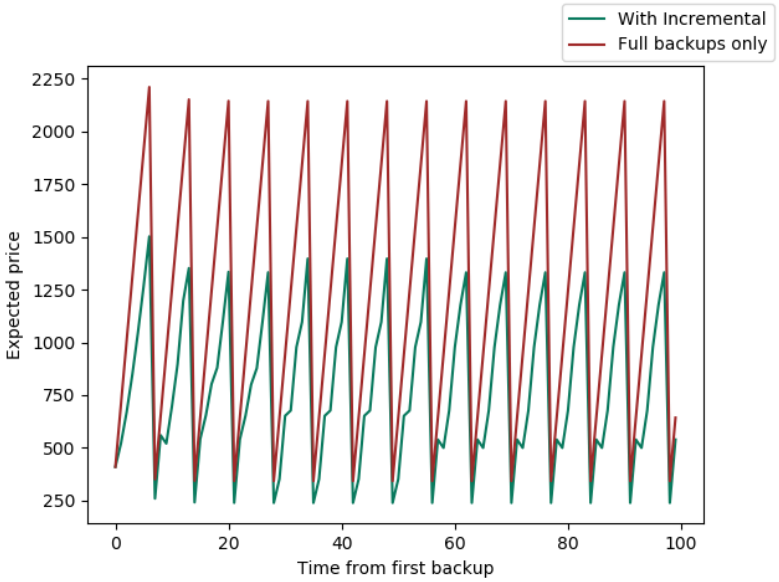
Комбинирайки уравнения 1 и 2 получаваме:

$$F(T) = p_F^k(W_T + k.r_F) + \sum_{i=0}^{k-1}(1-p_F).p_F^i\left(f(T_F) + (i+1).r_F\right) \quad (3)$$

Използвайки уравнения 1 и 3, можем да построим графика на очакваната цена с и без използването на инкрементални архиви.

[1] Можем да опитваме да възстановим само инкрементални архиви предхождащи следващият пълен архив

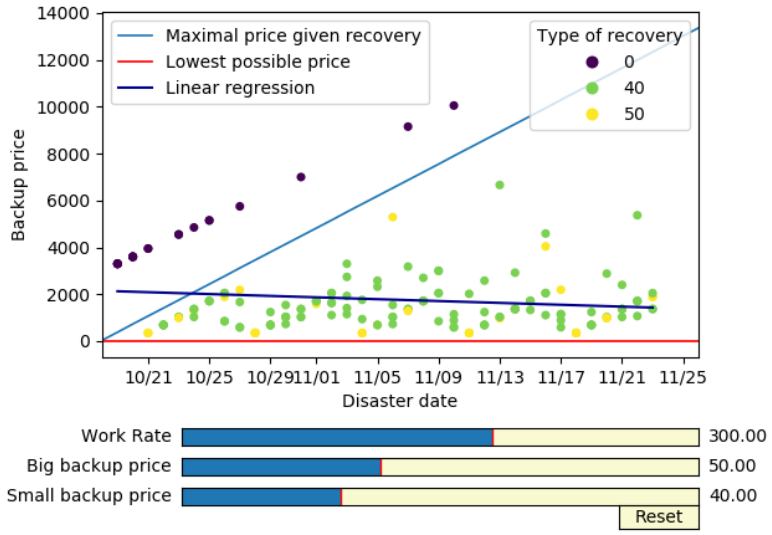
[2] Това съответства на $i-1$-вото събираемо в сумата от уравнение 3



Фигура 6: Full only and Whole model

4.6 Симулация Монте Карло

Направена бе симулация от тип Монте Карло на python, която генерира случайни процеси на възстановяване на данни с описаната структура на архивите. Цената на възстановяването беше направена на графика спрямо датата на атаката:



Фигура 7: Симулация Монте Карло

Цветовете във фигура 7 показват вида на последния успешно възстановен архив, пълен, инкрементален или несъществуващ.

Във фигура 6 и фигура 7 данните са за седмичен пълен архив и ежедневен инкрементален

Линейна регресия на данните беше генерирана, която показва как ефектът от начално неподсигурените данни намалява с времето, тъй като цената при провал се изчилсява като цената за преработването на всички данни, които компанията е генерирала.

4.7 Резултати

Беше построен модел за изчисляване на очакваната цена при възстановяване на данни. Нещо повече, ефектът от инкрементalните архиви беше показан в сравнение със стратегия използваща само пълни архиви. Беше направена и анализирана симулация от тип Монте Карло, коятодемонстрира реалния процес на възстановяване.

5 Бъдещо развитие

Авторът разглежда няколко посоки за бъдещото развитие на проакта, а именно:

- разглеждане на неконстантна скорост на работа за модела за архивиране
- разширяване на модела за откуп в посока описване на по сложни начини за разпространение на рансъмуера.
- използване на резултатите и базите данни на подобни проучвания с цел подкрепянето на модела с реални данни.[4]
- използване на динамичен модел за оценка на откупа

6 Благодарности

Искам да благодаря на своя ментор, Явор Папазов, и на Константин Делчев за безотказната помощ в избора на темата на проекта и последващото му развитие, за снабдяването ми с всички нужни материали за запознаването ми с темата, както и за изслушването на въпросите ми. Искам също да благодаря на Станислав Харизанов за професионалните съвети.

Литература

[1] Danny Yuxing Huang, Maxwell Matthaios Aliapoulios, Vector Guo Li, Luca Invernizzi, Elie Bursztein, Kylie McRoberts, Jonathan Levin, Kirill Levchenko, Alex C Snoeren, and Damon McCoy. Tracking ransomware end-to-end. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 618–631. IEEE, 2018.

[2] Tristan Caulfield, Christos Ioannidis, and David Pym. Dynamic pricing for ransomware.

[3] A Cartwright, Julio Hernandez-Castro, and Anna Stepanova. To pay or not: Game theoretic models of ransomware. In *Workshop on the Economics of Information Security (WEIS), Innsbruck, Austria*, 2018.

[4] Masarah Paquet-Clouston, Bernhard Haslhofer, and Benoit Dupont. Ransomware payments in the bitcoin ecosystem. *Journal of Cybersecurity*, 5(1):tyz003, 2019.

[5] Kurt Thomas, Danny Huang, David Wang, Elie Bursztein, Chris Grier, Thomas J Holt, Christopher Kruegel, Damon McCoy, Stefan Savage, and Giovanni Vigna. Framing dependencies introduced by underground commoditization. 2015.

[6] Amin Kharraz, William Robertson, Davide Balzarotti, Leyla Bilge, and Engin Kirda. Cutting the gordian knot: A look under the hood of ransomware attacks. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 3–24. Springer, 2015.

[7] J Michael Harrison, N Bora Keskin, and Assaf Zeevi. Bayesian dynamic pricing policies: Learning and earning under a binary prior distribution. *Management Science*, 58(3):570–586, 2012.

[8] Julio Hernandez-Castro, Edward Cartwright, and Anna Stepanova. Economic analysis of ransomware. *Available at SSRN 2937641*, 2017.

[9] Aron Laszka, Sadegh Farhang, and Jens Grossklags. On the economics of ransomware. In *International Conference on Decision and Game Theory for Security*, pages 397–417. Springer, 2017.

[10] Miguel Sousa Lobo and Stephen Boyd. Pricing and learning with uncertain demand. In *INFORMS Revenue Management Conference*, 2003.

[11] Michael Rothschild. A two-armed bandit theory of market pricing. *Journal of Economic Theory*, 9(2):185–202, 1974.

[12] S Nakamura, C Qian, S Fukumoto, and T Nakagawa. Optimal backup policy for a database system with incremental and full backups. *Mathematical and computer modelling*, 38(11-13):1373–1379, 2003.

[13] Cunhua Qian, Yingyan Huang, Xufeng Zhao, and Toshio Nakagawa. Optimal backup interval for a database system with full and periodic incremental backup. *JCP*, 5(4):557–564, 2010.