УС БАН '19

Рансъмуер и Гъвкави Стратегии за Бекъп

Автор: Никола Стайков

Ментор: Явор Папазов

1 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 бекъпи. 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

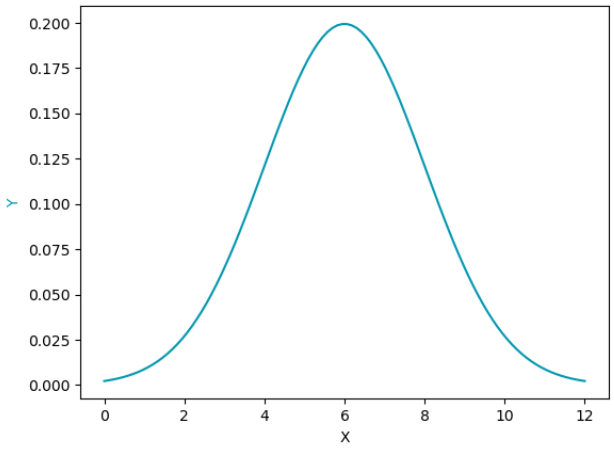
Този проект е разделен на две основни части, разглеждащи съответно модели за оптимизиране на откуп и оптимизиране на архивиране. Изследвайки обстойно двете противоположни позиции, можем да създадем пълна представа за стратегиите едновременно на атакуващите и жертвите. Обмисляйки начини как всяка от двете 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 откупа

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 бекъпи, 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 заключенията. Секцията е разделена на две смислови части, съответстващи на параметрите, които моделът изследва.

3.3.1 Размер на тестовата група и грешка

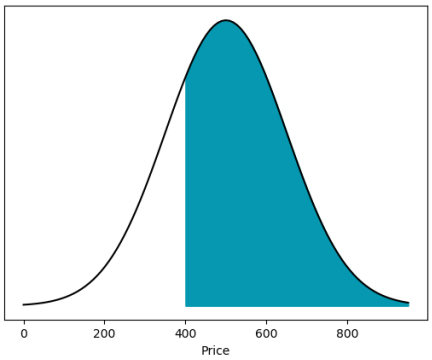
Тази секция описва математическия модел, използвам за оптимизиране на грешката. Изведени са заключения относно размера на тестовата група.

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

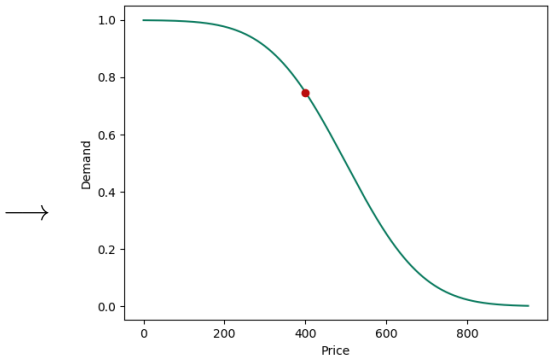
$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

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

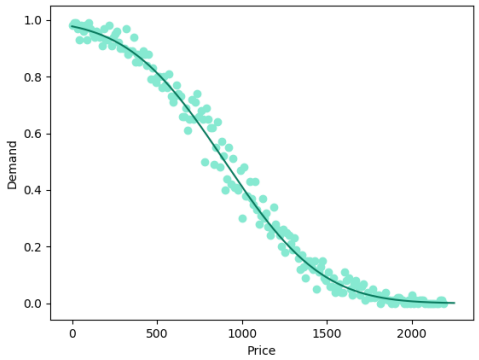


Фигура 2: Цена и търсене

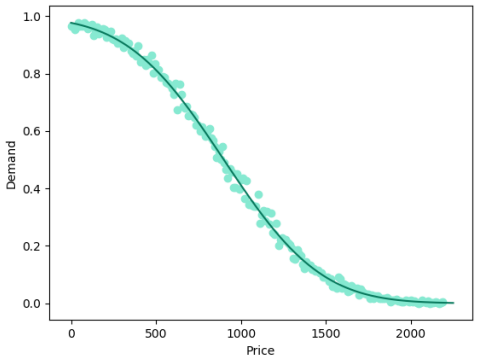
Отбелязаме, че интегралът трябва да бъде изчилсен до безкрайност, но след като k стигне $\mu + 3\sigma$, резултатът става пренебрежимо малък. Правейки това за цялата функция на разпределението получаваме кривата на търсенето чрез процента хора, които биха платили. Нека означим кривата на търсенето с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

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

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.

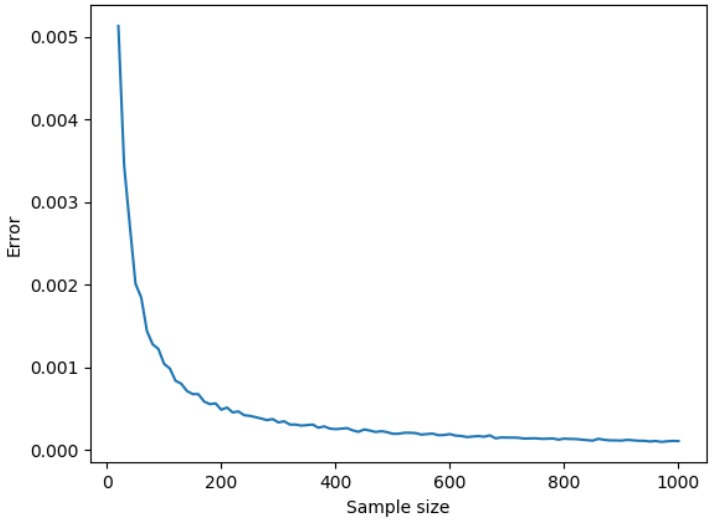


Фигура 3: 100 човека в групата



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

3.3.2 Бекъп функция

В тази секция описваме функция, отговаряща за вероятността за присъствието на бекъп. Изчислен е ефектът и върху очакваната печалба.

Първо нека дефинираме бекъп итератора b:

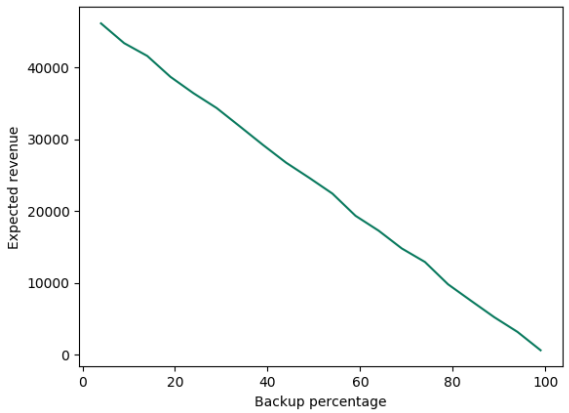
$$b = \begin{cases} 1 \text{ ако жертвата има бекъп,} \\ 0 \text{ ако жертвата няма бекъп} \end{cases}$$

Сега нека дефинираме желанието за плащане (ЖЗП):

$$P(x) = \begin{cases} d_x \text{ ако } b_i = 0, \\ c \text{ ако } b_i = 1 \end{cases}$$

Тук цената на бекъп е означена с c, а цената на данните на жертвата- с d_x.

Както и по-рано, изчисляваме очакваната вероятност даден човек да плати цена x. Приемаме, че вероятността конкретна жертва да има бекъп е константа p и изследваме как промяната на тази стойност влияе на очакваната печалба. Чрез събраните данни създаваме графика, която показва връзката между двете променливи. Измерената грешка е относителна.



3.4 Резултати

Изследвахме как размерът на тестовата група влияе на грешката на експерименталните данни и също как процента на защитените с бекъпи 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 бекъпи може да ни помогне да стигнем до подходящи подходи за справяне със заплахата.

4 Backup optimization

4.1 Introduction

When it comes to protection from ransomware, the most efficient method is building backups. This, however, has to be done regularly, as the effects after a potential attack will otherwise be insignificant. That is why it is important for backup protocols to be carefully build, considering both the risks of being attacked and the resources needed for the job. Furthermore, optimizing the backups can simultaneously increase the security level and save money.

In this section, a model for backing up data is considered in order to calculate the expected price. Such models, considering two options for backups: full and incremental, have been constructed and researched in the past[12][13]. We consider a cycle of backups, which repeats over between any two full backups and study how the intervals affect the expected price and how fast the effect of initial data, accumulated before the initial full backup, vanishes over time.

4.2 Theoretical setting

The idea behind the described model is to calculate and optimize the expected price of the recovery in case of an attack.

We will consider a backup as a structure, containing the following properties:

$$B \begin{cases} d\text{: the date on which the backup was made, as a day difference from a starting point} \\ p\text{: the probability that the recovery is unsuccessful for any reason} \\ r\text{: the price of trying to recover the data from the given backup} \end{cases}$$

Two types of backup will be considered:

1. Full backup: a backup of the whole database
2. Incremental backup: only saves the changes from the last backup

The backups from a certain type share common probability of failure and price for a recovery try.

In order for an incremental backup to be successful, all the incremental backups which precede it up to a full backup need to be successful as well as the full backup itself.

In this case data value should clearly be taken into account from a subjective point of view. Even though on the market some data may not be worth a lot, if it is essential for the functioning of a given company, it is clear that it will be willing to pay a lot to regain access to it immediately. Therefore, in the described model data value is considered as an ever-increasing amount, for the purposes of the research the "work rate namely the data value generated in a day, of the company is taken as a constant. We will denote it with w.

The cost of a backup recovery will be considered as a sum of two factors:

- The cost of redoing the lost work, denoted with W
- The cost of the recovery process itself, denoted with R

. We define $W = \Delta t.w$, where with Δt we denote the difference in days between the successful backup and the disaster date and $R = \sum_{i=1}^{n} r_i$, where the number of attempted backups is n and

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

Let the difference in days from the first backup to the disaster date be T. In case none of the backups are successful, we consider a variable W_T, corresponding to the price of redoing the whole work the company has done from the beginning. It is clear that $W_T > T.w$

4.3 Full backups only

When we only consider a set of full backups, the model is simply a Bernoulli distribution with finite trials, namely the number of full backups. We stop when we find a successful backup, starting from the latest and going to the last. Let us define the properties of a full backup(B_F):

$$B_F \begin{cases} p_F : \text{the probability of failure} \\ r_F : \text{the recovery trial cost} \\ t_F : \text{the days between two consecutive full backups} \end{cases}$$

Let k be the number of full backups made before the disaster date. Then:

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

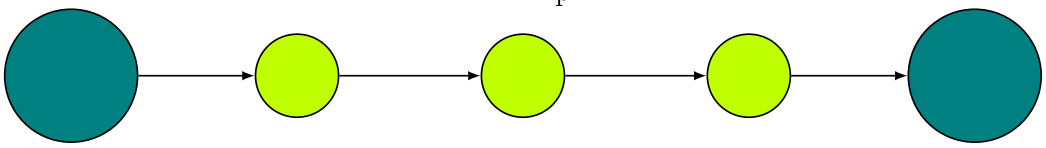
We can now define the expected backup cost:

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

This calculation is essential as incremental backups can only work when there is a working full backup and therefore the first thing we need to do is find the latest one. We can now move on to considering the incremental backups given a working full backup.

4.4 Incremental backups with a working full backup

We will now consider the case when we have a working backup and we are trying to recover additional data from the incremental backups.



Let us define the the properties of the incremental backup(B_I) in a similar fashion:

$$B_I \begin{cases} p_I : \text{the probability of failure} \\ r_I : \text{the recovery trial cost} \\ t_I : \text{the days between two consecutive incremental backups} \end{cases}$$

Let T_F denote the difference in days between the disaster date and the successful full backup and l denote the number of incremental backups we have to consider. We have two options for l depending on whether the latest full backup was successful:

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

Note that the last full backup being successful is equivalent to $T_F < t_F$.

We are in the exact opposite situation with respect to the previous subsection. The process of recovering incremental backups continues until we conduct an unsuccessful attempt to recover the data, as this will mean none of the following backups can be used either. Note that we are reducing W since in the initial position we are willing to redo the work up to the working full backup. That being said, we are ready to calculate the expected price:

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

Now we know how much the price will decrease when we use incremental backups and can build the whole picture using equations 1 and 2.

4.5 Overall expected price

For each summand in 1 we should add the effect of incremental backups, so we get new summands of the type:

$$P(W+R),$$

where P is the probability of a certain combination of events occurring, W is the cost of the data that has to be reworked and R is the cost of the recovery process. Incremental backups lower the cost of the data that has to be reworked but make R bigger. As mentioned before, there is only one case when the number of incremental backups we have to consider is different and it corresponds to the first full backup being successful. If the i-th full backup is successful[2]:

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

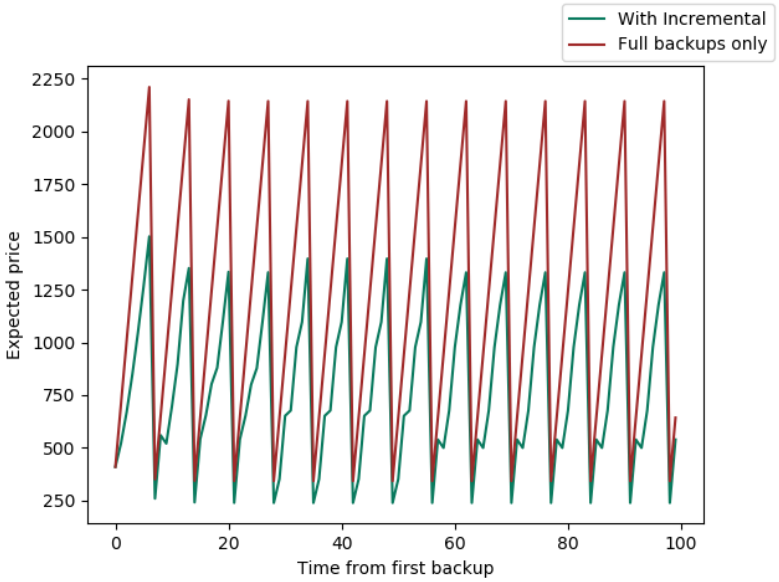
By combining equations 1 and 2 we get:

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

Using the described equations 1 and 3, we can construct a graph of the expected price with and without incremental backups included.

[1] We can only try to recover incremental backups preceding the next full backup

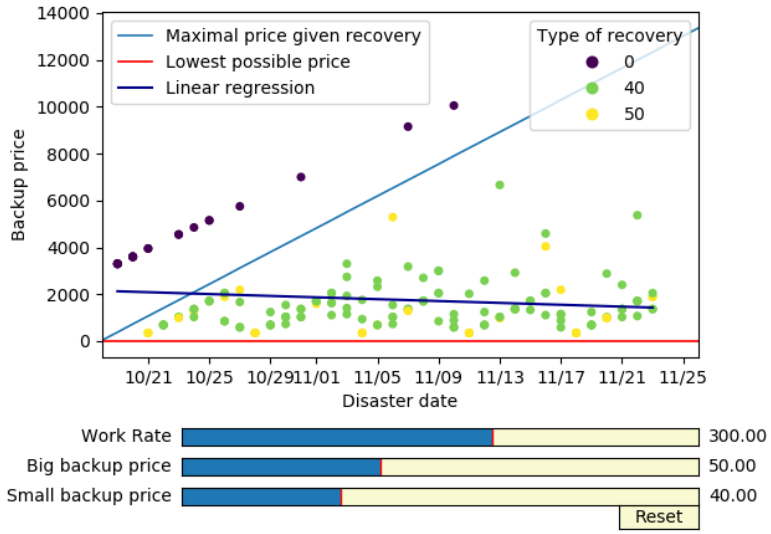
[2] This corresponds to the $i-1$-th summand in the sum from equation 3



Фигура 6: Full only and Whole model

4.6 Monte Carlo simulation

A Monte Carlo simulation has been build with python to generate random recovery processes with the described conditions of backup structure. The price of the recovery has been graphed with respect to the disaster date:



Фигура 7: Monte Carlo simulation

The colors in Figure 7 represent the type of the last backup, which was successful during the recovery, full, incremental or non-existing.

In both Figure 6 and Figure 7 the data showed is for a weekly full and daily incremental backups

A linear regression has been made of the data generated, which is to show that the effect of initial unsecured data fades with time, as the price of failure is calculated as the price to redo the whole work from the creation of the company.

4.7 Results

A model for backing up data has been built to calculate the expected price of backup recovery. Furthermore, the effect of incremental backups has been shown, as opposed to a strategy using only full backups. A Monte Carlo simulation has been built and analyzed to demonstrate the real process of recovery.

5 Further development

The author considers several future development directions for the project, namely:

- considering non-constant work rate for the backup model
- expanding the ransomware model to describe more complex way of distributing the ransomware
- using the results and databases of related studies in order to back the project with real data[4]
- considering a dynamic pricing model for the ransomware model

6 Бъдещо развитие

Авторът предвижда бъдещето развитие а проекта в няколко посоки, а именно:

- търсене на връзка между бекъпите и ЖЗП разпеделението
- разширяване на модела с цел да описва по-сложен начин на разпространение между машините в дадена система
- използването на разултатите и данните на други подобни разработки с цел подкрепянето на проекта с реални данни[4]
- разглеждане на динамичен модел за определяне на цената.

7 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.