Appendix to “Ransomware Activity and Blockchain Congestion”

This appendix provides additional material referenced in the paper. The appendix begins with explaining the details of the Common Vulnerability Scoring System (CVSS) and the rationale behind the choice of the variables reflecting ransomware activity. Next, it provides robustness checks. The appendix concludes with the discussion of the potential partial-equilibrium limit to transaction fees.

A1. Common vulnerability scoring system

Data feeds from the National Vulnerability Database (NVD) report a severity score of 1 to 10 for every vulnerability. This score is a function of the following exploitability factors:

$$Score = f(AccComp, Auth, AccVect, ConfImp, IntegImp, AvailImp),$$

where the inputs to the scoring function are:

- $AccComp$ is the vulnerability access complexity. High access complexity implies that a hacker needs to undertake many steps to exploit such vulnerability. A higher complexity lowers the overall score.
- $Auth$ is the number of times a hacker must pass authentication to a target to exploit a vulnerability. The higher this number, the lower the overall score.
- $AccVect$ is the means of access to the target. Some vulnerabilities require a hacker to have physical access to a target computer, while others can be exploited remotely. Remotely exploitable vulnerabilities receive a higher score *ceteris paribus*.
- $ConfImp$ is the extent to which the confidentiality of a target system is compromised. An increase in this metric increases the overall score.

- $IntegImp$ is the extent to which the integrity of a system is compromised. This metric captures whether a hacker can modify information on a target computer by exploiting the vulnerability. An increase in this metric increases the overall score.
- $AvailImp$ is the impact the attack may have on the availability of the target system. Specifically, it reflects the potential of a hacker attack to disrupt the target system. A higher availability impact increases the overall score.

Table A1 reports examples of released vulnerabilities. The vulnerability release date is recorded in Eastern Standard Time. In the main analysis, these dates are adjusted to match Bitcoin blockchain timestamps. Vulnerabilities with a high severity score typically allow a cybercriminal to take control of the target system and install malicious programs such as ransomware. For instance, the vulnerability CVE-2014-1776 has a severity score of 10.0 and can be used by remote attackers to execute arbitrary code on a local computer. This vulnerability appears on the Department of Homeland Security list of the top 30 most commonly exploited vulnerabilities.[1] Vulnerabilities scored around 5.0 do not usually allow ransomware attacks. For example, although attackers can exploit CVE-2015-6425 to cause a denial of service, this vulnerability does not allow a code execution on a target computer. Finally, vulnerabilities with a low severity score are difficult to exploit. For example, a cybercriminal must rely on help from a local user to take advantage of CVE-2014-4248.

[1] https://www.us-cert.gov/ncas/alerts/TA15-119A

A2. Severe vulnerabilities and ransomware activity

Figure A2 shows that there is an increase in the number of severe vulnerabilities around the day a new ransomware is listed in the Symantec dictionary.[2] The NVD data on vulnerabilities have several advantages over the ransomware list in the Symantec dictionary. First, a ransomware attack may not be successful, as it is the severity of the exploited vulnerability that defines the hacker's success. Second, older ransomware may be recycled to exploit newly discovered vulnerabilities. Third, Symantec does not always make a distinction between ransomware and trojans. Thus, I only use the NVD data on severe vulnerabilities in the main analysis.

A3. Ransomware-unrelated cybersecurity threats

The discovery of severe vulnerabilities may create shocks to demand for blockchain settlement through channels other than ransomware. For instance, coin theft risk may incentivize users to sell their coins and thus increase the number of blockchain transactions. To address this concern, I run the analysis excluding the days around the discovery of Bitcoin-related vulnerabilities and major coin theft episodes.

I obtain the release dates of Bitcoin-related vulnerabilities by screening vulnerability descriptions in the NVD database for the word "Bitcoin." Coin theft dates come from magoo.github.io/Blockchain-Graveyard, a website that traces major coin theft events. I remove three days before and after the dates of the coin theft episodes and Bitcoin-related vulnerability releases from the sample. In total, 71 non-overlapping days are removed. As shown in Table A2, the main results are not affected by elimination of the days surrounding coin thefts and Bitcoin-related vulnerability releases.

[2] https://www.symantec.com/security-center/a-z

A4. Stationarity

In this section, I address a concern about potential non-stationarity of the variables, as regressions on non-stationary data may produce spurious results. In what follows, I confirm that the model variables are stationary, mitigating spurious correlation concerns.

First, I test for stationarity of non-dummy independent variables in the regressions. $Vuln_t$ and the members of the $Controls_t$ vector are time series. As such, I rely on the conventional time series unit root tests to assess their stationarity. Specifically, I run augmented Dickey-Fuller and Phillips and Perron (1988) tests. The results in Table A3, Panel A, reject the unit root hypothesis with $p < 0.001$ for all independent variables.

Next, I test stationarity of the dependent panel variables: $nTrans_{i,t}$ and $TransFee_{i,t}$. Phillips and Moon (1999) show that although panels are unlikely to yield spurious outcomes, the issue is not eliminated completely. The rich literature on panel data stationarity proposes several panel unit root tests. These tests have different power depending on different panel dimensions and make various assumptions about the nature of the data-generating process (Maddala and Wu, 1999). I therefore take a prudent approach and run the five most common panel stationarity tests (Harris and Tzavalis, 1999; Maddala and Wu, 1999; Levin, Lin, and Chu, 2002; Im, Pesaran, and Shin, 2003; Breitung and Das, 2005). All of these tests reject the unit root hypothesis (Table A3, Panel B). As such, the results are unlikely to be driven by spurious correlation.

A5. A closer look into the relation between transaction fees and ransomware activity

When ransomware-unrelated users forego blockchain settlement in response to congestion, they mitigate transaction fee competition. As a result, time-constrained ransomware victims may refrain from attaching transaction fees that exceed the maximum fees that ransomware victims agree to pay. In this appendix, I test whether there exist a partial-equilibrium (pertaining to constraints of ransomware-unrelated users) limit to transaction fees. The limit is reached when a further increase in ransomware activity no longer increases the fees.

I start by sorting all sample days by the number of severe vulnerabilities released on those days and define the categories as follows: $Vuln3$ contains 30 days with the largest number of vulnerabilities released per day, $Vuln2$ contains days 31 to 60, and $Vuln1$ contains days 61 to 90.[3] Next, I estimate the following model:

$$TransFee_{i,t} = \alpha + \ \beta_1 Vuln1_t + \beta_2 Vuln2_t + \beta_3 Vuln3_t + Controls_t + \ \varepsilon_{i,t},$$

where $TransFee_{i,t}$ is the average transaction fee attached by wallet i on day t; $Vuln1_t - Vuln3_t$ are dummy variables equal to 1 if the observation falls into the corresponding category described in the paragraph above. $Controls_t$ is a vector of the following control variables: $RetBTC_t$ is the daily return on Bitcoin cryptocurrency; $RaisedICO_t$ is the sum of proceedings from the initial coin offering over the ten days surrounding an offering; $TrendBTC_t$ is the aggregate Google search volume for the word “Bitcoin” reported by trends.google.com; $FOMC_t$ is a dummy variable equal to one during the scheduled Federal Reserve Federal Open Market Committee meeting days and five days before and after such meetings.

Table A4 contains the coefficient estimates. Consistent with the earlier findings, the fees increase with the number of severe vulnerabilities. However, the coefficients β_2 and β_3 are statistically equal. As such, congestion typically increases the fees only to a certain limit. This suggests the possibility that there exists a partial-equilibrium limit to reward for mining. Importantly, this reasoning only reflects the constraints of ransomware-unrelated users.

[3] Adding further categories covering thirty days results in having multiple categories with the same number of vulnerabilities per day. These days are absorbed by the intercept term in the model.

References

Breitung, J. and Das, S., 2005, Panel Unit Root Tests under Cross-Sectional Dependence, Statistica Neerlandica 59, 414–433.

Harris, R. and Tzavalis, E., 1999, Inference for Unit Roots in Dynamic Panels where the Time Dimension is Fixed, Journal of Econometrics 91, 201-226.

Im, K., Pesaran, M., and Shin, Y., 2003, Testing for Unit Roots in Heterogeneous Panels, Journal of Econometrics 115, 53-74.

Levin, A., Lin, C., and Chu, C., 2002, Unit Root Tests in Panel Data: Asymptotic and Finite-Sample Properties, Journal of Econometrics 108, 1-24.

Maddala, G. and Wu, S., 1999, A Comparative Study of Unit Root Tests with Panel Data and a New Simple Test, Oxford Bulletin of Economics and Statistics 61(S1), 631-652.

Phillips, P. and Moon, H., 1999, Linear Regression Limit Theory for Nonstationary Panel Data, Econometrica 67, 1057-1111.

Phillips, P. and Perron, P., 1988, Testing for a Unit Root in Time Series Regression, Biometrika 75, 335-346.

Table A1: Vulnerability description and severity scores

ID	Description	Score	Date
CVE-2014-1776	Use-after-free vulnerability in Microsoft Internet Explorer 6 through 11 allows remote attackers to execute arbitrary code or cause a denial of service (memory corruption) via vectors related to the CMarkup: Is Connected to Primary Markup function, as exploited in the wild in April 2014.	10.0	04/27/2014
CVE-2015-6425	The WebApplications Identity Management subsystem in Cisco Unified Communications Manager 10.5 (0.98000.88) allows remote attackers to cause a denial of service (subsystem outage) via invalid session tokens, aka Bug ID CSCul83786.	5.0	12/16/2015
CVE-2014-4248	Unspecified vulnerability in the Oracle Application Object Library component in Oracle E-Business Suite 11.5.10.2, 12.0.6, 12.1.3, 12.2.2, and 12.2.3 allows local users to affect confidentiality via unknown vectors related to logging.	1.0	07/17/2014

Table A2: Ransomware-unrelated cybersecurity threats

This table examines whether the main results are sensitive to ransomware-unrelated cybersecurity risks. Coin theft risk may incentivize users to sell their coin holdings, thus increasing the number of blockchain transactions. To address this concern, I run the main analysis with the sample excluding three days before and after the discovery of Bitcoin-related vulnerabilities and major coin thefts. The specification is as follows:

$$DepVar_{,t} = \beta Vuln_t + Controls_t + \varepsilon_{i,t},$$

where $DepVar_{i,t}$ is either the number of Bitcoin blockchain transactions (Panel A) or the average transaction fee (Panel B) for wallet i on day t; $Vuln_t$ is the number of severe vulnerabilities. $Controls_t$ is a vector of the following control variables: $RetBTC_t$ is the daily return on Bitcoin cryptocurrency; $RaisedICO_t$ is the sum of proceedings from the initial coin offering over the ten days surrounding an offering; $TrendBTC_t$ is the aggregate Google search volume for the word "Bitcoin" reported by trends.google.com; $FOMC_t$ is a dummy variable equal to one during the scheduled Federal Reserve Federal Open Market Committee meeting days and five days before and after such meetings. Regressions control for individual Bitcoin wallet fixed effects and weekend fixed effects. Please refer to Section 3.1 in the paper for the definition of blockchain user Groups 1 to 4. All non-dummy variables are standardized. The standard errors (in parentheses) are clustered by wallet and day. Asterisks *** and ** denote statistical significance at the 1% and 5% levels.

	All	$Group1$	$Group2$	$Group3$	$Group4$
Panel A: Demand for Blockchain Settlement					
$Vuln$	.003***	-.005***	-.006***	-.006***	.005***
	(.000)	(.002)	(.001)	(.001)	(.000)
$WalletFE$	Yes	Yes	Yes	Yes	Yes
$YearQuarterFE$	Yes	Yes	Yes	Yes	Yes
$WeekendFE$	Yes	Yes	Yes	Yes	Yes
R^2	.0162	.0001	.0003	.0072	.0239
Panel B: Transaction Fees					
$Vuln$	.016***	-.008	.018**	.020***	.017***
	(.002)	(.010)	(.007)	(.006)	(.002)
$WalletFE$	Yes	Yes	Yes	Yes	Yes
$YearQuarterFE$	Yes	Yes	Yes	Yes	Yes
$WeekendFE$	Yes	Yes	Yes	Yes	Yes
R^2	.0008	.0003	.0024	.0038	.0014

Table A3: Stationarity

This table examines the potential non-stationarity of the sample variables. Panel A reports the results of augmented Dickey-Fuller and Phillips and Perron (1988) unit root tests for the time series variables, including $Vuln_t$, $RetBTC$, $RaisedICO_t$, and $TrendBTC_t$. Panel B reports the results of Harris and Tzavalis (1999), Maddala and Wu (1999), Levin, Lin, and Chu (2002), Im, Pesaran, and Shin (2003), and Breitung and Das (2005) unit root tests for the panel variables. Please refer to Tables 3 and 5 for the variable definitions. P-values are in parentheses. *** denotes statistical significance at the 1% level.

Panel A: Time series unit root tests		
	Augmented Dickey-Fuller	Phillips and Perron (1988)
H_0: $Vuln$ follows a unit root process	Rejected*** (.000)	Rejected*** (.000)
H_0: $RetBTC$ follows a unit root process	Rejected*** (.000)	Rejected*** (.000)
H_0: $RaisedICO$ follows a unit root process	Rejected*** (.000)	Rejected*** (.000)
H_0: $TrendBTC$ follows a unit root process	Rejected*** (.000)	Rejected*** (.000)

Panel B: Panel unit root tests					
	Harris and Tzavalis (1999)	Maddala and Wu (1999)	Levin, Lin, and Chu (2002)	Im, Pesaran, and Shin (2003)	Breitung and Das (2005)
H_0: $nTrans$ contains unit roots	Rejected*** (.000)	Rejected*** (.000)	Rejected*** (.000)	Rejected*** (.000)	Rejected*** (.000)
H_0: $TransFee$ contains unit roots	Rejected*** (.000)	Rejected*** (.000)	Rejected*** (.000)	Rejected*** (.000)	Rejected*** (.000)

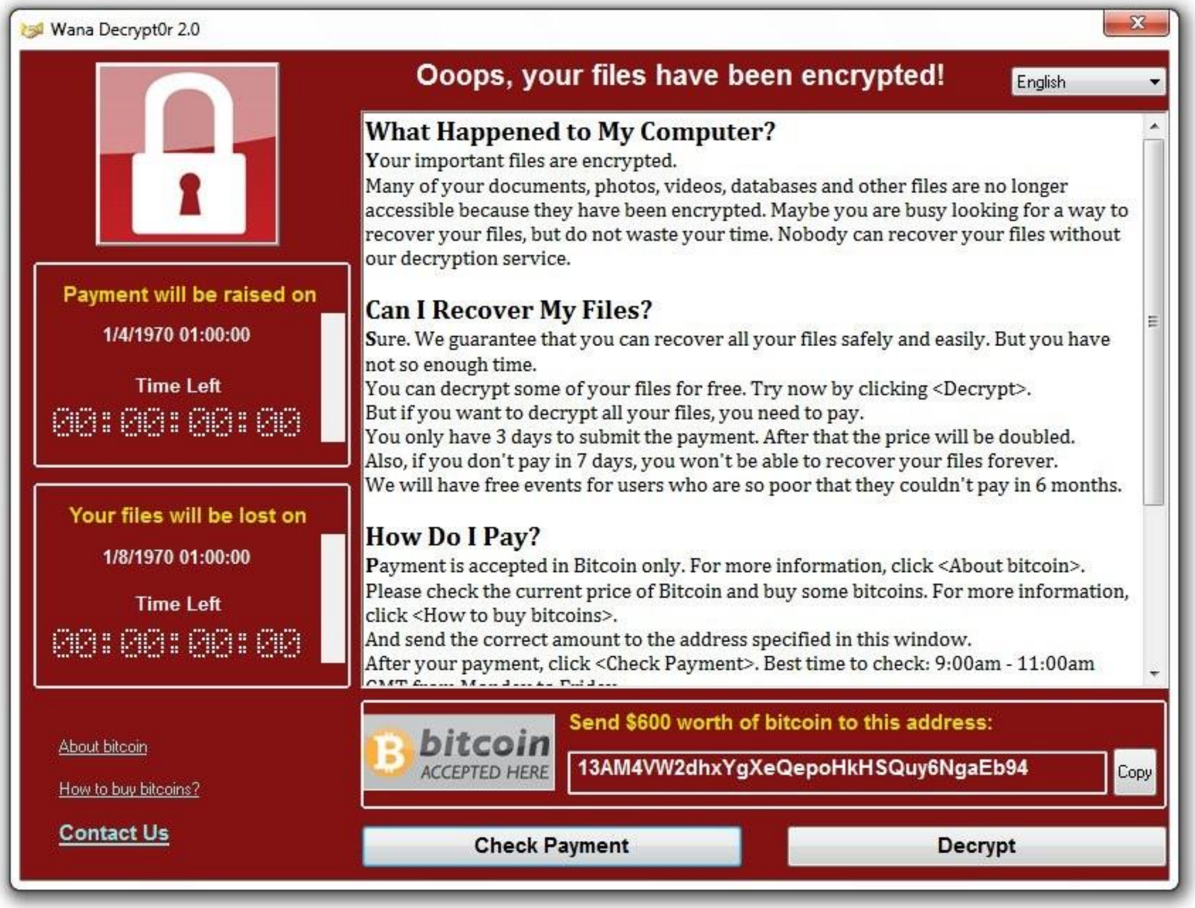
Table A4: Partial-equilibrium limit to transaction fees

This table tests whether there exist a partial-equilibrium (pertaining to constraints of ransomware-unrelated users) limit to transaction fees. I use the following model to find whether such a limit exists:

$$TransFee_{i,t} = \alpha + \beta_1 Vuln1_t + \beta_2 Vuln2_t + \beta_3 Vuln3_t + Controls_t + \varepsilon_{i,t},$$

where $TransFee_{i,t}$ is the average transaction fee attached by wallet i on day t; $Vuln3_t$ is a dummy variable equal to 1 if the number of vulnerabilities falls within the 30 days with the largest number of vulnerabilities released per day, $Vuln2_t$ contains days 31 to 60, and $Vuln1_t$ contains days 61 to 90. $Controls_t$ is a vector of the following control variables: $RetBTC_t$ is the daily return on Bitcoin cryptocurrency; $RaisedICO_t$ is the sum of proceedings from the initial coin offering over the ten days surrounding an offering; $TrendBTC_t$ is the aggregate Google search volume for the word "Bitcoin" reported by trends.google.com; $FOMC_t$ is a dummy variable equal to one during the scheduled Federal Reserve Federal Open Market Committee meeting days and five days before and after such meetings. Regressions control for individual Bitcoin wallet fixed effects, year and quarter fixed effects, and weekend fixed effects. All non-dummy variables are standardized. The standard errors (in parentheses) are clustered by wallet and day. Asterisks *** denote statistical significance at the 1% level.

$Vuln1$	-.016	H: $Vuln1 = Vuln2$	Rejected***
	(.014)	F-value	14.41
$Vuln2$	.043***	H: $Vuln1 = Vuln3$	Rejected***
	(.006)	F-value	12.72
$Vuln3$	.045***	H: $Vuln2 = Vuln3$	Not Rejected
	(.010)	F-value	.03
$Intercept$	-.026***		
	(.002)		
$RetBTC$	.012***		
	(.002)		
$RaisedICO$	.013**		
	(.005)		
$FOMC$	-.017***		
	(.002)		
$TrendBTC$	-.024***		
	(.003)		
$WalletFE$	Yes		
$YearQuarterFE$	Yes		
$WeekendFE$	Yes		
R^2	.0010		



Ooops, your files have been encrypted!

Payment will be raised on
1/4/1970 01:00:00
Time Left
00:00:00:00

Your files will be lost on
1/8/1970 01:00:00
Time Left
00:00:00:00

What Happened to My Computer?

Your important files are encrypted.
Many of your documents, photos, videos, databases and other files are no longer accessible because they have been encrypted. Maybe you are busy looking for a way to recover your files, but do not waste your time. Nobody can recover your files without our decryption service.

Can I Recover My Files?

Sure. We guarantee that you can recover all your files safely and easily. But you have not so enough time.
You can decrypt some of your files for free. Try now by clicking <Decrypt>.
But if you want to decrypt all your files, you need to pay.
You only have 3 days to submit the payment. After that the price will be doubled.
Also, if you don't pay in 7 days, you won't be able to recover your files forever.
We will have free events for users who are so poor that they couldn't pay in 6 months.

How Do I Pay?

Payment is accepted in Bitcoin only. For more information, click <About bitcoin>.
Please check the current price of Bitcoin and buy some bitcoins. For more information, click <How to buy bitcoins>.
And send the correct amount to the address specified in this window.
After your payment, click <Check Payment>. Best time to check: 9:00am - 11:00am GMT from Monday to Friday.

Send $600 worth of bitcoin to this address:
13AM4VW2dhxYgXeQepoHkHSQuy6NgaEb94

Figure A1: The WannaCry decryptor message

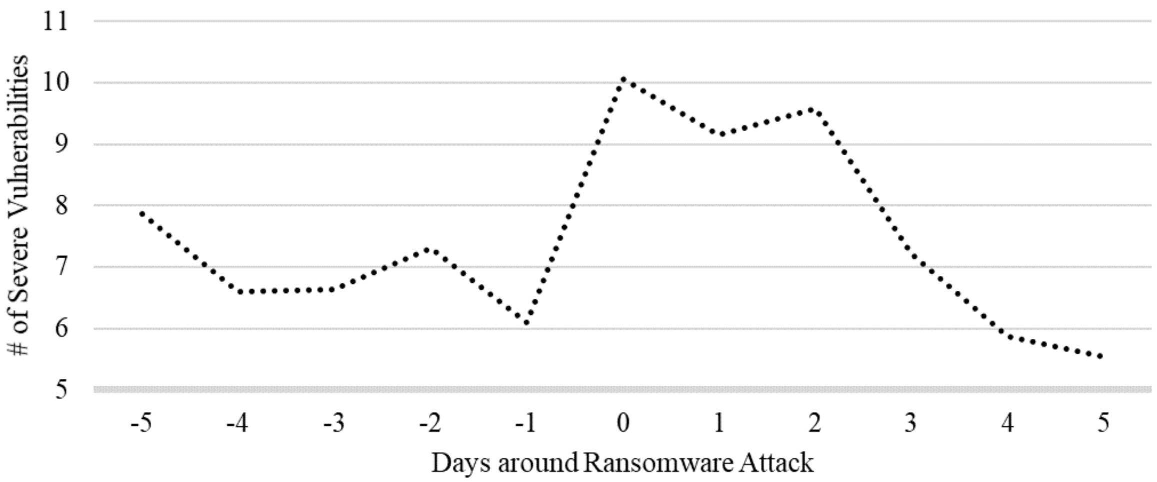


Figure A2: Severe vulnerabilities and ransomware releases