Appendix A
Proof of Theorem 1

Theorem 1 *The total size of files can be stored in FileInsurer is*

$$\min\left\{\frac{N_s\times minCapacity}{2r_1k},\frac{N_s\times minCapacity}{r_2}\right\},$$

where

$$r_1=\frac{\sum_f f.size\times f.value}{minValue\times\sum_f f.size}, \tag{1}$$

$$r_2=\frac{minCapcity\times\sum_f f.value}{minValue\times\sum_f f.size\times capPara}. \tag{2}$$

Proof: There are two restrictions on the total size of raw files. One is the restriction of total capacity. The other is the restriction of the maximal total value of stored files.

Under the former restriction, each file f is stored as $f.cp$ replicas. Due to the assumption of redundant capacity, The total size of all replicas can not exceed $\frac{1}{2}$ of total capacity. That is,

$$\sum_f (f.size\times f.cp)\le\frac{1}{2}N_s\times minCapacity.$$

Because $f.cp=k\times\frac{f.value}{minValue}$, we have

$$\sum_f (f.size\times k\times\frac{f.value}{minValue})\le\frac{1}{2}N_s\times minCapacity.$$

Then we have

$$k\times\frac{\sum_f (f.size\times f.value)}{minValue\times\sum_f f.size}\le\frac{1}{2}\times\frac{N_s\times minCapacity}{\sum_f f.size}.$$

Let $r_1=\frac{\sum_f f.size\times f.value}{minValue\times\sum_f f.size}$ we have

$$\frac{k}{r_1}\le\frac{1}{2}\times\frac{N_s\times minCapacity}{\sum_f f.size}.$$

Then

$$\sum_f f.size\le\frac{N_s\times minCapacity}{2r_1k}.$$

Under the latter restriction, the total value of files can't exceed $S_v^m\times minValue$. That is,

$$\sum_f f.value\le n\times minValue.$$

Because $n=Cap\_Para\times m$,

$$\frac{\sum_f f.value}{\sum_f f.size}\le\frac{Cap\_Para\times N_s\times minValue}{\sum_f f.size}.$$

Therefore,

$$\sum_f f.size\le\frac{N_s\times minCapacity}{r_2},$$

where

$$r_2=\frac{minCapcity\times\sum_f f.value}{minValue\times\sum_f f.size\times Cap\_Para}.$$

□

Appendix B
Proof of Theorem 2

Theorem 2 *If all files have the same size $f.size$, for a sector s with total capacity $s.capacity$ and free capacity $s.freeCap$, then*

$$\Pr\left[\exists s,\ s.freeCap\le\frac{1}{8}s.capacity\right]\le N_s\exp\left\{-0.144\frac{s.capacity}{f.size}\right\}.$$

Proof: In the special case, all files have the same size $f.size$. For a sector s with capacity $s.capacity$, it can store $\frac{s.capacity}{f.size}$ backups. We define $N_{cp}=kN_v$ as the number of file backups in total because $N_{cp}=\sum_f f.cp=\sum_f\frac{f.value}{minValue}\times k=kN_v$. Additionally, let X_i be the event that the backup i is stored in this sector and $S=\sum_{i=1}^{N_{cp}}X_i$. Because the assumption of redundant capacity, we have $\mathrm{E}[S]\le\frac{s.capacity}{2f.size}$. By multiplicative Chernoff bound, we have

$$\begin{aligned}&\Pr\left[s.freeCap\le\frac{1}{8}s.capacity\right]\\ =&\Pr\left[\sum_{i=1}^{N_{cp}}X_i\ge\frac{7}{8}\frac{s.capacity}{f.size}\right]\\ \le&\Pr\left[S\ge\frac{7}{4}\mathrm{E}[S]\right]\\ \le&\exp\left\{\left(\log\frac{e}{4}\right)\frac{3}{4}\mathrm{E}[S]\right\}\\ \le&\exp\left\{\left(\log\frac{e}{4}\right)\frac{3s.capacity}{8f.size}\right\}\\ \le&\exp\left\{-0.144\frac{s.capacity}{f.size}\right\}\end{aligned}$$

By applying union bound, we obtain

$$\Pr\left[\exists s,s.freeCap\le\frac{1}{8}s.capacity\right]\le N_s\exp\left\{-0.144\frac{s.capacity}{f.size}\right\}.$$

□

Appendix C
Proof of Theorem 3

We define the state of a FileInsurer network as (F,S,A,C) consisting of files F ,sectors S, all allocations A, and corrupted bits C in the network. Also, we define $V_{lost}^{(F,S,A,C)}$ as the sum of values of the lost files and $V_{confiscated}^{(F,S,A,C)}$ as the confiscated deposits of the corrupted sectors.

Lemma 1 *For a specific state (F, S, A, C), keeping the content and availability of each physical disk in the network unchanged, it can be viewed as another state (F',S,A',C) where the value of each file is $minValue$. State (F',S,A',C) satisfies $V_{lost}^{(F,S,A,C)}\le V_{lost}^{(F',S,A',C)}$.*

Proof: We divide each file descriptor f to $\frac{f.value}{minValue}$ different file descriptors. These file descriptors all have the value $minValue$ and the same Merkle root as that of f. Divide the $f.cp$ allocations of f equally among these new file descriptors, so each file descriptor have exactly k allocations. By defining F' as all these new file descriptors, A' as these new file allocations, we construct a state (F',S,A',C) such that the value of each file is $minValue$.

The content and availability of each physical disk in the network are same in state (F,S,A,C) and state (F',S,A',C). Since we do not change the state of sectors, we simply obtain $V_{confiscated}^{(F,S,A,C)}=V_{confiscated}^{(F',S,A',C)}$. For each file f lost in state (F,S,A,C), since all its backups are lost, every new file descriptor generated by f in state (F',S,A',C) also lost. The value of the file f is equal to the sum of the values of the file descriptors it generates, so $V_{lost}^{(F,S,A,C)}\le V_{lost}^{(F',S,A',C)}$. □

Lemma 2 *$\forall 0<p\le\frac{1}{5}$ and $5p\le x\le 1$, $D(x||p)\ge\frac{1}{2}x\log\frac{x}{p}$.*

Proof: In the proof below, we will use $x\ge p$ unspecified. Let

$$\begin{cases} f(x)=x^2\log\frac{x}{p}-(1-x)^2\log\frac{1-x}{1-p} \\ g(x)=\log\frac{x-1}{p-1}\left(-\log\frac{x}{p}+x-1\right)-x\log\frac{x}{p} \\ h(x)=\frac{x\log\frac{x}{p}}{-(1-x)\log\frac{1-x}{1-p}} \end{cases}$$

whose domain is $x\in[p,1]$. First, we have $f(x)\ge 0$ because $\frac{\mathrm{d}f}{\mathrm{d}x}=1+2D(x||p)\ge 0$ and $f(p)=0$. Then, we have $g(x)\ge 0$ because $\frac{\mathrm{d}g}{\mathrm{d}x}=\frac{f(x)}{x(1-x)}\ge 0$ and $g(p)=0$. Finally, we obtain $h(x)$ is monotonically increasing because $\frac{\mathrm{d}h}{\mathrm{d}x}=\frac{g(x)}{(x-1)^2\log^2\frac{1-x}{1-p}}\ge 0$.

Because $h(x)$ is monotonically increasing, $\forall x\ge 5p$, $h(x)\ge h(5p)=\frac{5p\log 5}{(1-5p)\log\frac{1-p}{1-5p}}$. We can find that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}p}\frac{5p\log 5}{(1-5p)\log\frac{1-p}{1-5p}}=&\frac{5\log 5\left((1-p)\log\frac{1-p}{1-5p}-4p\right)}{(1-5p)^2(1-p)\log^2\left(\frac{1-p}{1-5p}\right)}\\ \ge&\frac{5\log 5\left((1-p)(1-\frac{1-5p}{1-p})-4p\right)}{(1-5p)^2(1-p)\log^2\left(\frac{1-p}{1-5p}\right)}\\ =&0.\end{aligned}$$

When $p\to 0$, $\frac{5p\log 5}{(1-5p)\log\frac{1-p}{1-5p}}>2$, so $\forall 0<p\le\frac{1}{5}$, $\frac{5p\log 5}{(1-5p)\log\frac{1-p}{1-5p}}>2$, that is $\forall x\ge 5p, h(x)>2$. At last,

$$\begin{aligned}D(x||p)=&x\log\frac{x}{p}+(1-x)\log\frac{1-x}{1-p}\\ =&x\log\frac{x}{p}\left(1+\frac{(1-x)\log\frac{1-x}{1-p}}{x\log\frac{x}{p}}\right)\\ =&x\log\frac{x}{p}\left(1-\frac{1}{h(x)}\right)\\ \ge&\frac{1}{2}x\log\frac{x}{p}\end{aligned}$$

□

Lemma 3 *Assume that the total size of corrupted sectors is $\lambda N_s\times minCapacity$. Denote the total value of lost files to be V_{lost}. Then, with a probability of not less than $1-c$, V_{lost} satisfies*

$$V_{lost}\le minValue\times\max\left\{5N_v\lambda^k, N_v\lambda^{\frac{k}{2}}, 4\frac{\log\binom{N_s}{\lambda N_s}-\log c}{k\log\frac{1}{\lambda}}\right\}$$

Proof: Because of lemma 1, we can make the relaxation of that each file has value $minValue$ in subsequent analysis. Under the relaxations, the setting of the problem can be simplified as follow: there are N_v files, each file has the same value $minValue$ and needs to be stored in k sectors. Each storage location of each file is generated independent and identically distributed.

For any certain scheme that the adversary corrupts λ ratio of capacity, which means that the total size of corrupted sectors is $\lambda N_s\times minCapacity$, define random variable X_i as the indicator variable of that the file f_i is lost. Recall *storage randomness* indicates that all replicas are evenly and randomly distributed. Then all X_i are independent events and $\Pr[X_i]=\lambda^k$ when any sectors with total space of $\lambda N_s\times minCapacity$ are corrupted.

Denote $\gamma=\frac{V_{lost}^v}{minValue}$. By Chernoff bound and lemma 2, when $\gamma\ge 5N_v\lambda^k$, we obtain

$$\begin{aligned}&\Pr\left[\sum_i X_i\ge\gamma\right]\\ &\le\exp\left\{-N_v\left(\frac{\gamma}{N_v}\log\frac{\gamma}{N_v\lambda^k}+\left(1-\frac{\gamma}{N_v}\right)\log\frac{N_v-\gamma}{N_v-N_v\lambda^k}\right)\right\}\\ &\le\exp\left\{-\frac{\gamma}{2}\log\frac{\gamma}{N_v\lambda^k}\right\}\end{aligned}$$

For the adversary, it can corrupt λN_s sectors at will and wants to manufacture γ lost files. It has $\binom{N_s}{\lambda N_s}$ options to corrupt λN_s sectors.

Consider the number of scenarios in which an adversary can corrupt sectors with capacity $\lambda N_s\times minCapacity$. Here we do a simple scaling that treat a sector with $s.capacity$ capacity as $\frac{s.capacity}{minCapacity}$ sectors with capacity $minCapacity$. After this scaling, the adversary has $\binom{N_s}{\lambda N_s}$ options to corrupt sectors. Therefore, for the original situation, the number of scenarios in which an adversary can corrupt sectors with capacity $\lambda N_s\times minCapacity$ does not exceed $\binom{N_s}{\lambda N_s}$.

By union bound, the probability it cannot manufacture γ lost files is at least

$$1-\binom{N_s}{\lambda N_s}\exp\left\{-\frac{N_v\lambda^k\gamma\log\gamma}{2}\right\}.$$

Thus, when $\gamma\ge 5N_v\lambda^k$, we have the probability that an adversary cannot manufacture γ lost files is at least

$$1-\binom{N_s}{\lambda N_s}\exp\left\{-\frac{\gamma}{2}\log\frac{\gamma}{N_v\lambda^k}\right\}.$$

As $\gamma\ge N_v\lambda^{\frac{k}{2}}$, $\log\frac{\gamma}{N_v\lambda^k}\ge\log\left(\lambda^{\frac{-k}{2}}\right)=-\frac{k}{2}\log\lambda$.

$$\begin{aligned}&\gamma\ge 4\frac{\log\binom{N_s}{\lambda N_s}-\log c}{k\log\frac{1}{\lambda}}\\ \Leftrightarrow&\gamma\frac{k}{4}\log\frac{1}{\lambda}\ge-\log\frac{c}{\binom{N_s}{\lambda N_s}}\\ \Rightarrow&\frac{\gamma}{2}\log\frac{\gamma}{N_v\lambda^k}\ge-\log\frac{c}{\binom{N_s}{\lambda N_s}}\\ \Leftrightarrow&-\frac{\gamma}{2}\log\frac{\gamma}{N_v\lambda^k}\le\log\frac{c}{\binom{N_s}{\lambda N_s}}\\ \Leftrightarrow&\exp\left\{-\frac{\gamma}{2}\log\frac{\gamma}{N_v\lambda^k}\right\}\le\frac{c}{\binom{N_s}{\lambda N_s}}\\ \Leftrightarrow&\binom{N_s}{\lambda N_s}\exp\left\{-\frac{\gamma}{2}\log\frac{\gamma}{N_v\lambda^k}\right\}\le c\end{aligned}$$

Then with a probability of no less than $1-c$, γ satisfies

$$\gamma\le\max\left\{5\lambda^k N_v, \lambda^{\frac{k}{2}}N_v, 4\frac{\log\binom{N_s}{\lambda N_s}-\log c}{k\log\frac{1}{\lambda}}\right\}$$

Therefore, with a probability of no less than $1-c$, V_{lost}^v satisfies

$$V_{lost}^v\le minValue\times\max\left\{5N_v\lambda^k, N_v\lambda^{\frac{k}{2}}, 4\frac{\log\binom{N_s}{\lambda N_s}-\log c}{k\log\frac{1}{\lambda}}\right\}$$

□

Theorem 3 *Assume that the total size of corrupted sectors is $\lambda N_s\times minCapacity$. Denote the total value of lost files to be V_{lost}, and $\gamma_{lost}^v=\frac{V_{lost}}{S_v\times minValue}$ represents the ratio of the* *value of lost files to the total value of all files. Then with a probability of not less than $1-c$, γ_{lost}^{v} satisfies*

$$\gamma_{lost}^{v}\le\max\left\{5\lambda^{k},\lambda^{\frac{k}{2}},\frac{4\left(\frac{\log\frac{e}{2\pi}-\log c}{N_s}-\log\left(\lambda^{\lambda}(1-\lambda)^{1-\lambda}\right)\right)}{\gamma_{v}^{m}k\log\frac{1}{\lambda}\times capPara}\right\}.$$

Proof: Now we use an upper bound of the binomial number via Stirling's formula,

$$\begin{aligned}\binom{N_s}{\lambda N_s}&=\frac{N_s!}{(\lambda N_s)!(N_s-\lambda N_s)!}\\&\le\frac{e}{2\pi}\frac{N_s^{N_s+\frac{1}{2}}}{(\lambda N_s)^{\lambda N_s+\frac{1}{2}}(N_s-\lambda N_s)^{N_s-\lambda N_s+\frac{1}{2}}}\\&=\frac{e}{2\pi}\sqrt{\frac{1}{N_s\lambda(1-\lambda)}}\left(\frac{1}{\lambda^{\lambda}(1-\lambda)^{1-\lambda}}\right)^{N_s}\\&\le\frac{e}{2\pi}\left(\frac{1}{\lambda^{\lambda}(1-\lambda)^{1-\lambda}}\right)^{N_s}\end{aligned}$$

Using this upper bound, we can have a simpler version of γ_{lost}^{v}:

$$\begin{aligned}\gamma_{lost}^{v}&\le\max\left\{5\lambda^{k},\lambda^{\frac{k}{2}},4\frac{\log\frac{e}{2\pi}-N_s\log\left(\lambda^{\lambda}(1-\lambda)^{1-\lambda}\right)-\log c}{N_v k\log\frac{1}{\lambda}}\right\}\\&=\max\left\{5\lambda^{k},\lambda^{\frac{k}{2}},4\frac{\frac{\log\frac{e}{2\pi}-\log c}{N_s}-\log\left(\lambda^{\lambda}(1-\lambda)^{1-\lambda}\right)}{capPara\cdot\gamma_{v}^{m}k\log\frac{1}{\lambda}}\right\}\end{aligned}$$

□

Appendix D
Proof of Theorem 4

Theorem 4 *Assume that the total size of corrupted sectors is no more than $\lambda N_s\times minCapacity$. If the deposit ratio satisfies*

$$\gamma_{deposit}\ge\max\left\{5\lambda^{k-1},\lambda^{\frac{k}{2}-1},\frac{4}{k\times capPara}\left(\frac{\log N_s}{\log\frac{1}{\lambda}}+\frac{\log\frac{1}{c}}{\log N_s}\right)\right\},$$

then full compensation can be achieved with a probability of not less than $1-c$.

Proof: By assumption, the total size of corrupted sectors is no more than $\lambda N_s\times minCapacity$. Because the deposit of corrupted sectors should always cover the file loss, $\forall\frac{1}{N_s}\le\lambda'\le\lambda$ we shall have $\lambda'\gamma_{deposit}N_v^m\ge\gamma$, which is equivalent to

$$\gamma_{deposit}\ge\max_{\frac{1}{N_s}\le\lambda'\le\lambda}\left\{\frac{\gamma}{\lambda' N_v^m}\right\}.$$

Then with probability no less than $1-c$, the following $\gamma_{deposit}$ is enough for full compensation

$$\gamma_{deposit}\ge\max_{\frac{1}{N_s}\le\lambda'\le\lambda}\max\left\{5(\lambda')^{k-1},(\lambda')^{\frac{k}{2}-1},4\frac{\log\binom{N_s}{\lambda' N_s}-\log c}{\lambda' N_v^m k\log\frac{1}{\lambda'}}\right\}.$$

Considering the third part, we have

$$\begin{aligned}&4\frac{\log\binom{N_s}{\lambda' N_s}-\log c}{\lambda' N_v^m k\log\frac{1}{\lambda'}}\\&\le\max_{\frac{1}{N_s}\le\lambda'\le\lambda}\frac{4\log\left(N_s^{\lambda' N_s}\right)-4\log c}{\lambda' N_v^m k\log\frac{1}{\lambda'}}\\&=\max_{\frac{1}{N_s}\le\lambda'\le\lambda}\frac{4\lambda' N_s\log N_s-4\log c}{\lambda' N_v^m k\log\frac{1}{\lambda'}}\\&\le\left(\max_{\frac{1}{N_s}\le\lambda'\le\lambda}\frac{4N_s\log N_s}{N_v^m k\log\frac{1}{\lambda'}}\right)+\left(\max_{\frac{1}{N_s}\le\lambda'\le\lambda}\frac{-4\log c}{\lambda' N_v^m k\log\frac{1}{\lambda'}}\right)\\&\le\frac{4N_s\log N_s}{N_v^m k\log\frac{1}{\lambda}}+\frac{-4N_s\log c}{N_v^m k\log N_s}.\end{aligned}$$

Then

$$\gamma_{deposit}\ge\max\left\{5\lambda^{k-1},\lambda^{\frac{k}{2}-1},\frac{4N_s\log N_s}{N_v^m k\log\frac{1}{\lambda}}+\frac{-4N_s\log c}{N_v^m k\log N_s}\right\}.$$

As $capPara=\frac{N_v^m}{N_s}$,

$$\gamma_{deposit}\ge\max\left\{5\lambda^{k-1},\lambda^{\frac{k}{2}-1},\frac{4}{k\times capPara}\left(\frac{\log N_s}{\log\frac{1}{\lambda}}+\frac{\log\frac{1}{c}}{\log N_s}\right)\right\}.$$

□

Appendix E
experiment

Recall that $N_{cp}=kN_v$ is the number of file backups and each file f needs to store $f.cp$ backups on the network. Table IV shows the result of the capacity usage of the most loaded sector in the experiment. We run the experiment under two different settings. In the first setting, we reallocate all file backups in one go for 100 times. In the second setting, we allocate each file backup and then randomly refresh the location of a file backup $100N_{cp}$ times. In the experiment, we record the maximum value of capacity usage at any time. The results are shown in table IV. That the maximum capacity usage is less than 1 means that no file backups are allocated to sectors with insufficient capacity. We can find that under the distributions we test in the experiments, the probability of that file backups are allocated to sectors with insufficient capacity is sufficiently small.

TABLE IV
EXPERIMENT RESULT: MAXIMUM CAPACITY USAGE OF SECTORS

reallocate all file backups 100 times						
parameter		maximum capacity usage				
N_{cp}	N_s	[1]	[2]	[3]	[4]	[5]
10^5	5	0.511	0.508	0.514	0.511	0.509
10^5	10	0.519	0.518	0.521	0.518	0.515
10^5	20	0.525	0.524	0.536	0.530	0.529
10^5	50	0.565	0.539	0.558	0.549	0.548
10^5	100	0.571	0.566	0.584	0.572	0.569
10^6	50	0.515	0.513	0.517	0.515	0.513
10^6	100	0.522	0.523	0.530	0.530	0.521
10^6	200	0.538	0.530	0.542	0.534	0.533
10^6	500	0.558	0.548	0.569	0.570	0.557
10^6	1000	0.591	0.571	0.598	0.594	0.576
10^7	500	0.516	0.515	0.522	0.521	0.518
10^7	1000	0.524	0.521	0.531	0.528	0.524
10^7	2000	0.540	0.534	0.544	0.545	0.534
10^7	5000	0.562	0.554	0.581	0.573	0.560
10^7	10000	0.589	0.576	0.609	0.606	0.585
10^8	5000	0.520	0.518	0.522	0.520	0.517
10^8	10000	0.526	0.525	0.537	0.529	0.524
10^8	20000	0.541	0.534	0.550	0.547	0.538
10^8	50000	0.562	0.555	0.580	0.571	0.559
10^8	10^5	0.591	0.582	0.614	0.599	0.586

refresh the location of a file backup $100N_{cp}$ times						
parameter		maximum capacity usage				
N_{cp}	N_s	[1]	[2]	[3]	[4]	[5]
10^5	5	0.517	0.511	0.519	0.515	0.514
10^5	10	0.524	0.523	0.529	0.522	0.519
10^5	20	0.532	0.529	0.538	0.535	0.531
10^5	50	0.550	0.551	0.566	0.554	0.557
10^5	100	0.588	0.571	0.599	0.595	0.581
10^6	50	0.518	0.516	0.521	0.519	0.517
10^6	100	0.525	0.522	0.532	0.529	0.526
10^6	200	0.536	0.535	0.546	0.542	0.541
10^6	500	0.565	0.563	0.582	0.575	0.562
10^6	1000	0.592	0.581	0.610	0.605	0.589
10^7	500	0.520	0.518	0.525	0.523	0.522
10^7	1000	0.533	0.527	0.534	0.533	0.531
10^7	2000	0.542	0.535	0.553	0.549	0.540
10^7	5000	0.565	0.562	0.586	0.582	0.569
10^7	10000	0.610	0.591	0.626	0.613	0.599
10^8	5000	0.529	0.527	0.539	0.532	0.529
10^8	10000	0.542	0.536	0.543	0.546	0.537
10^8	20000	0.551	0.547	0.560	0.558	0.548
10^8	50000	0.575	0.569	0.599	0.584	0.577
10^8	10^5	0.611	0.604	0.639	0.628	0.611

[1]: Uniform distribution in interval [0,1]
[2]: Uniform distribution in interval [1,2]
[3]: Exponential distribution
[4]: Normal distribution with $\mu = \sigma^2$
[5]: Normal distribution with $\mu = 2\sigma^2$