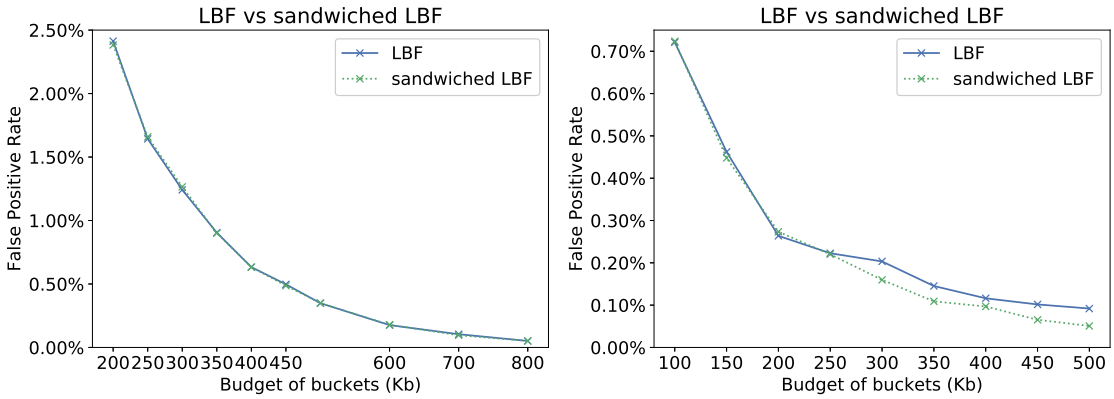
A More comparisons between the LBF and sandwiched LBF



(a) (b)

Figure A.1: FPR comparison between LBF and sandwiched LBF under different bitmap sizes. (a) malicious URL experiment; (b) malware detection experiment

B Comparing the Bloom filter to Hierarchical Hashing

The machine learning model used in the learned Bloom filters is critical because it has discrimination power between the keys and non-keys and is more efficient in identifying keys in some cases. To show its unique role, we replaced the machine learning model with another Bloom filter such that it becomes a hierarchical Bloom filter (learner is replaced by an initial filter). To implement the hierarchical Bloom filter, we spare 50% of the bit budget to the initial filter and use the other bits to build the backup filter.

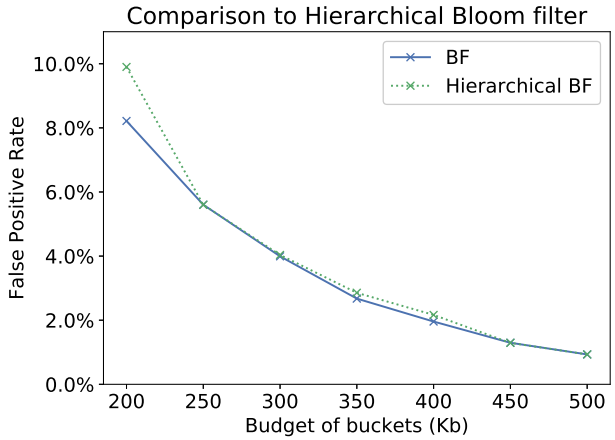


Figure B.1: FPR comparison between LBF and sandwiched LBF under different bitmap sizes. (a) malicious URL experiment; (b) malware detection experiment

Figure B.1 shows that the hierarchical BF does not outperform the original BF under all the budget of

buckets, and in some cases, it even achieves a worse FPR. Hence, using a random hash function to replace the learner is not a memory efficient approach.

C Proof of the Statements

Proof of Lemma 1:

Proof. Let $Z_j(x) = \sum_{i=1}^{m} \mathbb{1}(s(x) \in [\tau_{j-1}, \tau_j) | x \notin S)$, then $Z_j(x) \sim Bernoulli(p_j)$, and $m_j = \sum_{i=1}^{m} Z_j(x_i)$ counts the number of non-keys falling in group j and $\hat{p}_j = \frac{m_j}{m}$. To upper bound the probability of the overall estimation error of p_j, first, we need to evaluate its expectation, $\mathbb{E}\left(\sum_{j=1}^{K} |\hat{p}_j - p_j|\right)$.

Since m_j is a binomial random variable, its exact cdf is hard to compute. But with central limit theorem, when m is large, $\frac{m_j - mp_j}{\sqrt{mp_j(1-p_j)}} \longrightarrow N(0,1)$. Thus, we can approximate $\mathbb{E}\left(|\hat{p}_j - p_j|\right) = \mathbb{E}\left(|\frac{m_j - mp_j}{\sqrt{mp_j(1-p_j)}}|\right) \cdot \sqrt{\frac{p_j(1-p_j)}{m}} \approx \sqrt{\frac{2}{\pi}} \cdot \sqrt{\frac{p_j(1-p_j)}{m}}$ (if $Z \sim N(0,1)$, $\mathbb{E}\left(|Z|\right) = \sqrt{\frac{2}{\pi}}$). Then, the expectation of overall error is approximated by $\mathbb{E}\left(\sum_{j=1}^{K} |\hat{p}_j - p_j|\right) \approx \sqrt{\frac{2}{m\pi}} \cdot \left(\sum_{j=1}^{K} \sqrt{p_j(1-p_j)}\right)$, which goes to 0 as m becomes larger.

We need to further upper bound the tail probability of $\sum_{j=1}^{K} |\hat{p}_j - p_j|$. First, we upper bound the variance of $\sum_{j=1}^{K} |\hat{p}_j - p_j|$,

$$\begin{aligned}
\mathrm{Var}\left(\sum_{j=1}^{K} |\hat{p}_j - p_j|\right) &\leq K \sum_{j=1}^{K} \mathrm{Var}\left(|\hat{p}_j - p_j|\right) \\
&= K \sum_{j=1}^{K} \left(\mathrm{Var}\left(\hat{p}_j - p_j\right) - \mathbb{E}\left(|\hat{p}_j - p_j|\right)^2\right) \\
&\approx \frac{K}{m} \sum_{j=1}^{K} \left(p_j(1-p_j) - \frac{2}{\pi}\left(\sum_{i=1}^{K} \sqrt{p_j(1-p_j)}\right)^2\right) \triangleq \frac{K}{m} V(\mathbf{p})
\end{aligned}$$

Now, by envoking the Chebyshev's inequality,

$$\begin{aligned}
& \mathrm{P}\left[\sum_{j=1}^{K} |\hat{p}_j - p_j| \geq \epsilon\right] \\
= \; & \mathrm{P}\left[\sum_{j=1}^{K} |\hat{p}_j - p_j| - \mathbb{E}\left(\sum_{j=1}^{K} |\hat{p}_j - p_j|\right) \geq \epsilon - \mathbb{E}\left(\sum_{j=1}^{K} |\hat{p}_j - p_j|\right)\right] \\
\leq \; & \frac{\mathrm{Var}\left(\sum_{j=1}^{K} |\hat{p}_j - p_j|\right)}{\left(\epsilon - \mathbb{E}\left(\sum_{j=1}^{K} |\hat{p}_j - p_j|\right)\right)^2} \\
= \; & \frac{KV(\mathbf{p})}{m\left(\epsilon - \mathbb{E}\left(\sum_{j=1}^{K} |\hat{p}_j - p_j|\right)\right)^2} \longrightarrow 0 \text{ as } m \longrightarrow \infty
\end{aligned}$$

Thus, $\sum_{j=1}^{K}|\hat{p}_j - p_j|$ converges to 0 in probability as $m \longrightarrow \infty$. $\square$

Moreover, since we have

$$\begin{aligned}\mathbb{E}\left(\sum_{j=1}^{K}|\hat{p}_j - p_j|\right) &\approx \sqrt{\frac{2}{m\pi}}(\sum_{j=1}^{K}\sqrt{p_j(1-p_j)}) \\ &\leq \sqrt{\frac{2}{m\pi}(K-1)} \end{aligned} \tag{1}$$

$$\begin{aligned} V(\mathbf{p}) &= \sum_{j=1}^{K}\left(p_j(1-p_j) - \frac{2}{\pi}\left(\sum_{i=1}^{K}\sqrt{p_j(1-p_j)}\right)^2\right) \\ &\leq \sum_{j=1}^{K}\left(p_j(1-p_j)\left(1-\frac{2}{\pi}\right)\right) \\ &\leq \left(1-\frac{2}{\pi}\right)\left(1-\frac{1}{K}\right) \end{aligned} \tag{2}$$

Then, by Eq 1 and Eq 2, we can upper bound $\mathrm{P}\left[\sum_{j=1}^{K}|\hat{p}_j - p_j| \geq \epsilon\right]$ by,

$$\begin{aligned}\mathrm{P}\left[\sum_{j=1}^{K}|\hat{p}_j - p_j| \geq \epsilon\right] &\leq \frac{KV(\mathbf{p})}{m\left(\epsilon - \mathbb{E}\left(\sum_{j=1}^{K}|\hat{p}_j - p_j|\right)\right)^2} \\ &\leq \frac{(1-\frac{2}{\pi})(K-1)}{m\left(\epsilon - \sqrt{\frac{2}{m\pi}(K-1)}\right)^2} \end{aligned} \tag{3}$$

When $m \geq \frac{2(k-1)}{\epsilon^2}\left[\sqrt{\frac{1}{\pi}} + \sqrt{\frac{1-2/\pi}{\delta}}\right]^2$, we have $m\left(\epsilon - \sqrt{\frac{2}{m\pi}(K-1)}\right)^2 \geq \frac{(K-1)(1-\frac{2}{\pi})}{\delta}$, thus, $\mathrm{P}\left[\sum_{j=1}^{K}|\hat{p}_j - p_j| \geq \epsilon\right] \leq \delta$. $\square$

Proof of Theorem 1:

Proof. For comparison, we choose $\tau = \tau_{g-1}$, for both LBF and Ada-BF, queries with scores larger than τ are identified as keys directly by the same machine learning model. Thus, to compare the overall FPR, we only need to evaluate the FPR of queries with score lower than τ.

Let $p_0 = \mathrm{P}\left[s(x) < \tau | x \notin S\right]$ be the probability of a key with score lower than τ. Let n_0 denote the number of keys with score less than τ, $n_0 = \sum_{i:x_i \in S} I(s(x_i) < \tau)$. For learned Bloom filter using K hash functions, the

expected FPR follows,

$$\begin{aligned}\mathbb{E}\left(\text{FPR}\right) &= (1-p_0)+p_0\left(1-\left(1-\frac{1}{R}\right)^{Kn_0}\right)^K \\ &= 1-p_0+p_0\beta^K, \end{aligned} \tag{4}$$

where R is the length of the Bloom filter. For Ada-BF, assume we fix the number of groups g. Then, we only need to determine K_{max} and $K_{min} = K_{max} - g + 1$. Let $p_j = Pr(\tau_{j-1} \leq s(x) < \tau_j | x \notin S)$ The expected FPR of the Ada-BF is,

$$\begin{aligned}\mathbb{E}\left(\text{FPR}_a\right) &= \sum_{j=1}^{g} p_j \left(1-\left(1-\frac{1}{R}\right)^{\sum_{j=1}^{g-1} K_j n_j}\right)_j^K \\ &= \sum_{j=1}^{g-1} p_j \alpha^{K_j}, \end{aligned} \tag{5}$$

where $\sum_{j=1}^{g-1} n_j = n_0$. Next, we give a strategy to select K_{max} which ensures a lower FPR of Ada-BF than LBF.

Select $K_{max} = \lfloor K + \frac{g}{2} - 1 \rfloor$. Let $T_i = n_i - n_{i-1}$. Then, we have

$$\begin{aligned} n_0 K = \sum_{j=1}^{g-1} n_j K &= K\left[n_1 + \sum_{i=2}^{g-1}(n_1 + \sum_{i=1}^{j-1} T_i)\right] \\ &= K\left[n_1(g-1) + \sum_{j=1}^{g-2} T_j(g-j-1)\right] \\ &= Kn_1(g-1) + \frac{2K}{g-2}\left[\sum_{j=1}^{g-2}\frac{(g-2)(g-1-j)}{2}T_j\right] \\ &\leq \frac{2K}{g-2}\left[\frac{(g-1)(g-2)}{2}n_1 + \sum_{j=1}^{g-2}\frac{(g+j-2)(g-1-j)}{2}T_j\right] \\ &\overset{(1)}{\leq} \frac{2K}{g-2}\sum_{j=1}^{g-1}(j-1)n_j, \end{aligned} \tag{6}$$

where inequality (1) comes from $\sum_{j=1}^{g-2}\frac{(g-2)(g-1-j)}{2}T_j \leq \sum_{j=1}^{g-2}\frac{(g+j-2)(g-1-j)}{2}T_j$. By Eq 6. we further get the relationship between α and β.

$$\begin{aligned} \sum_{j=1}^{g-1} K_j n_j &= \sum_{j=1}^{g-1}(K_{max} - j + 1)n_j \\ &\leq n_0\left(K_{max} - \frac{g}{2} + 1\right) \leq n_0 K \\ &\implies \alpha \leq \beta. \end{aligned}$$

Moreover, since in the main text, we have seen,

$$\begin{aligned}\mathbb{E}\left(\text{FPR}\right) &= \sum_{j=1}^{g} p_j \alpha^{K_j} = \frac{\sum_{j=1}^{g} c^{g-j}\alpha^{K_j}}{\sum_{j=1}^{g} c^{g-j}} \\ &\leq \begin{cases} \dfrac{(1-c)(1-(c\alpha)^g)}{(\frac{1}{\alpha}-c)(\alpha^g-(c\alpha)^g)}\alpha^{K_{max}}, & c\alpha \neq 1 \\ \dfrac{1-c}{1-c^g}\cdot g, & c\alpha = 1 \end{cases}\end{aligned}$$

Thus,

$$\begin{aligned}\mathbb{E}\left(\text{FPR}_a\right) &= \frac{(1-c)(1-(c\alpha)^g)}{(\frac{1}{\alpha}-c)(\alpha^g-(c\alpha)^g)}\alpha^{K_{max}} \\ &\leq \frac{(1-c)(1-(c\alpha)^g)}{(\frac{1}{\alpha}-c)(\alpha^g-(c\alpha)^g)}\beta^{K_{max}} \\ &\leq \beta^{K_{max}}\frac{\alpha(c-1)}{c\alpha-1} \\ &< \mathbb{E}\left(\text{FPR}\right)\left(\frac{1+\lambda}{\lambda}\beta^{K_{max}-K}\right) \\ &\leq \mathbb{E}\left(\text{FPR}\right)\left(\frac{1+\lambda}{\lambda}\beta^{\lfloor g/2-1\rfloor}\right).\end{aligned}$$

Therefore, as g increases, the upper bound of $\mathbb{E}\left(\text{FPR}_a\right)$ decreases exponentially fast. Moreover, since $\frac{1+\lambda}{\lambda}$ is a constant, when g is large enough, we have $\frac{1+\lambda}{\lambda}\beta^{\lfloor g/2-1\rfloor} \leq 1$. Thus, the $\mathbb{E}\left(\text{FPR}_e\right)$ is reduced to strictly lower than $\mathbb{E}\left(\text{FPR}\right)$. □

Proof of Theorem 2:

Proof. Let $\eta = \frac{\log(c)}{\log(\mu)} \approx \frac{\log(c)}{\log(0.618)} < 0$. By the tuning strategy described in the previous section, we require the expected false positive items should be similar across the groups. Thus, we have

$$\begin{aligned} & p_1 \cdot \mu^{R_1/n_1} = p_j \cdot \mu^{R_j/n_j} \\ \Longrightarrow \quad & R_j = n_j\left(\frac{R_1}{n_1} + (j-1)\eta\right), \text{ for } j \in [g-1]\end{aligned}$$

where R_j is the budget of buckets for group j. For group j, since all the queries are identified as keys by the machine learning model directly, thus, $R_g = 0$. Given length of Bloom filter for group 1, R_1, the total budget of buckets can be expressed as,

$$\sum_{j=1}^{g-1} R_j = \sum_{j=1}^{g-1}\frac{n_j}{n_1}R_1 + (j-1)n_j\eta$$

Let $p_0 = Pr(s(x) < \tau | x \notin S)$ and $p_j = Pr(\tau_{j-1} \leq s(x) < \tau_j | x \notin S)$. Let n_0 denote the number of keys with score less than τ, $n_0 = \sum_{i:x_i \in S} I(s(x_i) < \tau)$, and n_j be the number of keys in group j, $n_j = \sum_{i:x_i \in S} I(\tau_{j-1} \leq s(x_i) < \tau_j)$. Due to $\tau = \tau_{g-1}$, we have $\sum_{j=1}^{g-1} n_j = n_0$. Moreover, since $\tau_{g-1} = \tau$, queries with score higher than τ have the same FPR for both disjoint Ada-BF and LBF. So, we only need to compare the FPR of the two methods when the score is lower than τ. If LBF and Ada-BF achieve the same optimal expected FPR, we have

$$\begin{aligned} p_0 \cdot \mu^{R/n_0} &= \sum_{j=1}^{g-1} p_j \cdot \mu^{R_j/n_j} = g \cdot p_1 \cdot \mu^{R_1/n_1} \\ \implies R &= \frac{n_0}{n_1} R_1 - n_0 \frac{\log(p_0/p_1) - \log(g)}{log(\mu)} \\ &= \sum_{j=1}^{g-1} \left[\frac{n_j}{n_1} R_1 - n_j \frac{\log(1 - \left(\frac{1}{c}\right))^g - \log\left(1 - \frac{1}{c}\right) - \log(g)}{\log(\mu)} \right], \end{aligned}$$

where R is the budget of buckets of LBF. Let $T_j = n_{j+1} - n_j \geq 0$. Next, we upper bound $\sum_{j=1}^{g-1} n_j$ with $\sum_{j=1}^{g-1}(j-1)n_j$.

$$\begin{aligned} \sum_{j=1}^{g-1} n_j &= n_1 + \sum_{i=2}^{g-1}(n_1 + \sum_{i=1}^{j-1} T_i) \\ &= n_1(g-1) + \sum_{j=1}^{g-2} T_j(g-j-1) \\ &= \frac{2}{g-2} \left[\frac{(g-1)(g-2)n_1}{2} + \sum_{j=1}^{g-2} \frac{(g-2)(g-1-j)T_j}{2} \right] \\ &\leq \frac{2}{g-2} \left[\frac{(g-1)(g-2)n_1}{2} + \sum_{j=1}^{g-2} \frac{(g+j-2)(g-1-j)T_j}{2} \right] \\ &= \frac{2}{g-2} \sum_{j=1}^{g-1} (j-1)n_j \end{aligned}$$

Therefore, we can lower bound R,

$$R \geq \sum_{j=1}^{g-1} \left[\frac{n_j}{n_1} R_1 - (j-1)n_j \frac{2(\log(1 - \left(\frac{1}{c}\right))^g - \log\left(1 - \frac{1}{c}\right) - \log(g))}{(g-2)\log(\mu)} \right].$$

Now, we can lower bound $R - \sum_{j=1}^{g-1} R_j$,

$$R - \sum_{j=1}^{g-1} R_j \geq \sum_{j=1}^{g-1} (j-1)n_j \left[-\eta - \frac{2(\log(1 - \left(\frac{1}{c}\right))^g - \log\left(1 - \frac{1}{c}\right) - \log(g))}{(g-2)\log(\mu)} \right].$$

Since η is a negative constant, while $\frac{2(\log(1-\left(\frac{1}{c}\right))^g - \log\left(1-\frac{1}{c}\right) - \log(g))}{(g-2)\log(\mu)}$ approaches to 0 when g is large. Therefore, when g is large, $\eta - \frac{2(\log(1-\left(\frac{1}{c}\right))^g - \log\left(1-\frac{1}{c}\right) - \log(g))}{(g-2)\log(\mu)} < 0$ and $R - \sum_{j=1}^{g-1} R_j$ is strictly larger than 0. So, disjoint Ada-BF consumes less memory than LBF to achieve the same expected FPR. □