One Line Factoring and Partial Key Exposure Attack on RSA
Description of the algorithm

Step 1 Compute $s = \lceil \sqrt{n} \rceil$.
Step 2 Compute $m = s^2 \mod n$.
Step 3 Check whether $m$ is perfect square.
   - If yes compute $t = \sqrt{n}$. Then $gcd(s - t, n) = p$.
   - If not increase $i$ and go to Step 1.

Running time: $O(n^{1/3} + \epsilon)$. 
Description of the algorithm

Step 1 Compute \( s = \lceil \sqrt{n} \rceil \).

Step 2 Compute \( m = s^2 \mod n \).

Step 3 Check whether \( m \) is perfect square.
- If yes compute \( t = \sqrt{n} \). Then \( \gcd(s - t, n) = p \).
- If not increase \( i \) and go to Step 1.

Running time: \( O(n^{1/3} + \epsilon) \).
Assume that $n = p \cdot q$ has prime factors $n^{1/3} < p, q < n^{2/3}$. 
Assume that $n = p \cdot q$ has prime factors $n^{1/3} < p, q < n^{2/3}$. We search for a solution to
Assume that $n = p \cdot q$ has prime factors $n^{1/3} < p, q < n^{2/3}$. We search for a solution to

$$y^2 = \left(\lceil \sqrt[n]{ni} \rceil \right)^2 - ni$$

by iterating $i$ and looking for squares after reduction modulo $n$. 
Suppose \( ni \) is not a perfect square.
Analysis of the algorithm

Suppose $ni$ is not a perfect square. Then $ni = u^2 + a$ where $0 < a < 2u + 1$. 
Analysis of the algorithm

Suppose \( ni \) is not a perfect square. Then \( ni = u^2 + a \) where \( 0 < a < 2u + 1 \). Since \( ni = u^2 + a > u^2 \) we have that \( \sqrt{ni} > u \) and so
Analysis of the algorithm

Suppose \( ni \) is not a perfect square. Then \( ni = u^2 + a \) where \( 0 < a < 2u + 1 \). Since \( ni = u^2 + a > u^2 \) we have that \( \sqrt{ni} > u \) and so \( \lceil \sqrt{ni} \rceil = u + 1 \). This implies that
Analysis of the algorithm

Suppose \( ni \) is not a perfect square. Then \( ni = u^2 + a \) where \( 0 < a < 2u + 1 \). Since \( ni = u^2 + a > u^2 \) we have that \( \sqrt{ni} > u \) and so \( \lceil \sqrt{ni} \rceil = u + 1 \). This implies that

\[
(\lceil \sqrt{ni} \rceil)^2 - ni = (u + 1)^2 - (u^2 + a) = 2u + 1 - a.
\]
Analysis of the algorithm

Suppose $ni$ is not a perfect square. Then $ni = u^2 + a$ where $0 < a < 2u + 1$. Since $ni = u^2 + a > u^2$ we have that $\sqrt{ni} > u$ and so $\lceil \sqrt{ni} \rceil = u + 1$. This implies that

$$(\lceil \sqrt{ni} \rceil)^2 - ni = (u + 1)^2 - (u^2 + a) = 2u + 1 - a.$$ 

Let $s = (\lceil \sqrt{ni} \rceil)$. Then the equation above can be written as

$$(s^2 - ni) = 2u + 1 - a.$$
Suppose $ni$ is not a perfect square. Then $ni = u^2 + a$ where $0 < a < 2u + 1$. Since $ni = u^2 + a > u^2$ we have that $\sqrt{ni} > u$ and so $\lceil \sqrt{ni} \rceil = u + 1$. This implies that

$$(\lceil \sqrt{ni} \rceil)^2 - ni = (u + 1)^2 - (u^2 + a) = 2u + 1 - a.$$ 

Let $s = (\lceil \sqrt{ni} \rceil)$. Then the equation above can be written as

$$(s^2 - ni) = 2u + 1 - a.$$ 

Then the equation

$$(s^2 - ni) \mod n = 2u + 1 - a \mod n$$

is equivalent to
Suppose \( ni \) is not a perfect square. Then \( ni = u^2 + a \) where \( 0 < a < 2u + 1 \). Since \( ni = u^2 + a > u^2 \) we have that \( \sqrt{ni} > u \) and so \( \lceil \sqrt{ni} \rceil = u + 1 \). This implies that

\[
(\lceil \sqrt{ni} \rceil)^2 - ni = (u + 1)^2 - (u^2 + a) = 2u + 1 - a.
\]

Let \( s = (\lceil \sqrt{ni} \rceil) \). Then the equation above can be written as

\[
(s^2 - ni) = 2u + 1 - a.
\]

Then the equation

\[
(s^2 - ni) \mod n = 2u + 1 - a \mod n
\]

is equivalent to

\[
s^2 \mod n = (2u + 1 - a) \mod n.
\]
Let $s^2 \mod n = m$. 
Let $s^2 \mod n = m$. We search for values for which $m$ is a square i.e. for an integer $x$ such that $x^2 = m$. When we find such an $x$ we have
Let $s^2 \mod n = m$. We search for values for which $m$ is a square i.e. for an integer $x$ such that $x^2 = m$. When we find such an $x$ we have

$$x^2 = s^2 \mod n$$
Let \( s^2 \mod n = m \). We search for values for which \( m \) is a square i.e. for an integer \( x \) such that \( x^2 = m \). When we find such an \( x \) we have

\[
x^2 = s^2 \mod n
\]

which implies that either \( n | (x - s) \) or \( n | (x + s) \) and that either \( \gcd(x - s, n) \) or \( \gcd(x + s, n) \) gives a non-trivial factor of \( n \).
Analysis of the algorithm (cont.)

Note that since $0 < a < 2u = 1$ and $u < \sqrt{ni}$ we have that
Note that since $0 < a < 2u = 1$ and $u < \sqrt{ni}$ we have that

$$m = s^2 \mod n \leq 2u + 1 - 1 = 2u < 2\sqrt{ni}$$
Note that since $0 < a < 2u = 1$ and $u < \sqrt{ni}$ we have that

$$m = s^2 \mod n \leq 2u + 1 - 1 = 2u < 2\sqrt{ni}$$

This implies that there are approximately $\sqrt{2}(ni)^{\frac{1}{4}}$ squares less than $2\sqrt{ni}$. So the probability that we can find such an $x$ is $\frac{1}{\sqrt{2}(ni)^{\frac{1}{4}}}$. 

Assume that we do $n^{\frac{1}{3}}$ iterations. Then the probability to find $x$ is $n^{\frac{1}{3}} \sqrt{2}(ni)^{\frac{1}{4}} = \frac{1}{\sqrt{2}} > 0.5$. Using Fermat's method we will need $\sqrt{n} \sqrt{2} > n^{\frac{1}{3}}$ iterations to find a square $x$ with probability greater than 0.5.
Note that since $0 < a < 2u = 1$ and $u < \sqrt{ni}$ we have that

$$m = s^2 \mod n \leq 2u + 1 - 1 = 2u < 2\sqrt{ni}$$

This implies that there are approximately $\sqrt{2(\sqrt{2}(ni)^{1/4}}$ squares less than $2\sqrt{ni}$. So the probability that we can find such an $x$ is

$$\frac{1}{\sqrt{2}(ni)^{1/4}}.$$ 

Assume that we do $n^{1/3}$ iterations. Then the probability to find $x$ is
Note that since \(0 < a < 2u = 1\) and \(u < \sqrt{ni}\) we have that

\[m = s^2 \mod n \leq 2u + 1 - 1 = 2u < 2\sqrt{ni}\]

This implies that there are approximately \(\sqrt{2(ni)^{1/4}}\) squares less than \(2\sqrt{ni}\). So the probability that we can find such an \(x\) is \(\frac{1}{\sqrt{2(ni)^{1/4}}}\).

Assume that we do \(n^{1/3}\) iterations. Then the probability to find \(x\) is

\[\frac{n^{1/3}}{\sqrt{2(ni)^{1/4}}} = \frac{1}{\sqrt{2}} > 0.5\]
Note that since $0 < a < 2u = 1$ and $u < \sqrt{ni}$ we have that

$$m = s^2 \mod n \leq 2u + 1 - 1 = 2u < 2\sqrt{ni}$$

This implies that there are approximately $\sqrt{2}(ni)^{\frac{1}{4}}$ squares less than $2\sqrt{ni}$. So the probability that we can find such an $x$ is $\frac{1}{\sqrt{2}(ni)^{\frac{1}{4}}}$. Assume that we do $n^{\frac{1}{3}}$ iterations. Then the probability to find $x$ is

$$\frac{n^{\frac{1}{3}}}{\sqrt{2}(ni)^{\frac{1}{4}}} = \frac{1}{\sqrt{2}} > 0.5$$

Using Fermat’s method we will need $\frac{\sqrt{n}}{\sqrt{2}} > n^{\frac{1}{3}}$ iterations to find a square $x$ with probability greater than 0.5.
Note that since $0 < a < 2u = 1$ and $u < \sqrt{ni}$ we have that

$$m = s^2 \mod n \leq 2u + 1 - 1 = 2u < 2\sqrt{ni}$$

This implies that there are approximately $\sqrt{2(ni)^\frac{1}{4}}$ squares less than $2\sqrt{ni}$. So the probability that we can find such an $x$ is $\frac{1}{\sqrt{2}(ni)^\frac{1}{4}}$.

Assume that we do $n^{\frac{1}{3}}$ iterations. Then the probability to find $x$ is

$$\frac{n^{\frac{1}{3}}}{\sqrt{2}(ni)^{\frac{1}{4}}} = \frac{1}{\sqrt{2}} > 0.5$$

Using Fermat’s method we will need $\frac{\sqrt{n}}{\sqrt{2}} > n^{\frac{1}{3}}$ iterations to find a square $x$ with probability greater than 0.5.
Scenario

Two partners in a business decide that neither will have access to the private key \( d \) of the business, but instead they will each have a share of the key. This is to ensure that either can read encrypted confidential information only with the collaboration of the other. If one of them gets the least significant half of \( d \), and the other gets the most significant half of \( d \), then the one holding the least significant half has a distinct advantage in reconstructing the whole private key - especially if the prime numbers \( p \) and \( q \) have the same number of digits.
Recall that

\[ ed \ mod \ \phi(n) = 1 \]

This implies that there is a
Recall that

$$ed \mod \phi(n) = 1$$

This implies that there is a $k < \min\{e, d\}$ such that
Recall that

\[ ed \mod \phi(n) = 1 \]

This implies that there is a \( k < \min\{e, d\} \) such that

\[ ed = 1 + k\phi(n) \]
Recall that

\[ ed \mod \phi(n) = 1 \]

This implies that there is a \( k < \min\{e, d\} \) such that

\[ ed = 1 + k\phi(n) \]

From the point of view of the outsider, \( k \) and \( \phi(n) \) are unknown, but \( n \) and \( e \) are known.
Recall that

\[ ed \mod \phi(n) = 1 \]

This implies that there is a \( k < \min\{e, d\} \) such that

\[ ed = 1 + k\phi(n) \]

From the point of view of the outsider, \( k \) and \( \phi(n) \) are unknown, but \( n \) and \( e \) are known. However, the **AM-GM** inequality

\[ 2\sqrt{xy} \leq x + y \]

can be used to approximate \( \phi(n) \).
Recall that

\[ ed \mod \phi(n) = 1 \]

This implies that there is a \( k < min\{e, d\} \) such that

\[ ed = 1 + k\phi(n) \]

From the point of view of the outsider, \( k \) and \( \phi(n) \) are unknown, but \( n \) and \( e \) are known. However, the AM-GM inequality

\[ 2\sqrt{xy} \leq x + y \]

can be used to approximate \( \phi(n) \).

\[ \phi(n) = (p - 1)(q - 1) = \]
Recall that

\[ ed \mod \phi(n) = 1 \]

This implies that there is a \( k < \min\{e, d\} \) such that

\[ ed = 1 + k\phi(n) \]

From the point of view of the outsider, \( k \) and \( \phi(n) \) are unknown, but \( n \) and \( e \) are known. However, the **AM-GM** inequality

\[ 2\sqrt{xy} \leq x + y \]

can be used to approximate \( \phi(n) \).

\[ \phi(n) = (p - 1)(q - 1) = pq - (p + q) + 1 \leq pq - 2\sqrt{pq} + 1 \]
Recall that
\[ ed \ mod \ \phi(n) = 1 \]

This implies that there is a \( k < \min\{e, d\} \) such that
\[ ed = 1 + k\phi(n) \]

From the point of view of the outsider, \( k \) and \( \phi(n) \) are unknown, but \( n \) and \( e \) are known. However, the AM-GM inequality
\[ 2\sqrt{xy} \leq x + y \]

can be used to approximate \( \phi(n) \).
\[
\phi(n) = (p - 1)(q - 1) = pq - (p + q) + 1 \leq pq - 2\sqrt{pq} + 1 = n - 2\sqrt{n} + 1
\]
For $k < d$ define

$$D_k = \frac{k(n-2\sqrt{n}+1)+1}{e}.$$
For $k < d$ define

$$D_k = \frac{k(n-2\sqrt{n}+1)+1}{e}.$$ 

Then $D_k$ is the attacker’s first approximation of the private key $d$. 

For $k < d$ define

$$D_k = \frac{k(n-2\sqrt{n}+1)+1}{e}.$$ 

Then $D_k$ is the attacker’s first approximation of the private key $d$. If the attacker also has the least significant digits of $d$, say $L$, then the attacker can improve the approximation $D_k$ by replacing the corresponding least significant digits of $D_k$ by the digits of $L$. 
For $k < d$ define

$$D_k = \frac{k(n-2\sqrt{n}+1)+1}{e}.$$ 

Then $D_k$ is the attacker’s first approximation of the private key $d$. If the attacker also has the least significant digits of $d$, say $L$, then the attacker can improve the approximation $D_k$ by replacing the corresponding least significant digits of $D_k$ by the digits of $L$. Let the resulting improvement be $d_k$. 
Problem

How good an approximation for $d$ is $d_k$?

Lemma

$$|d_k - d| = \left(\frac{k}{e}\right)(\sqrt{p} - \sqrt{q})^2.$$
Problem

How good an approximation for $d$ is $d_k$?

Lemma

$$|d_k - d| = \left(\frac{k}{e}\right)(\sqrt{p} - \sqrt{q})^2.$$  

Since $k < e$, the following inequalities hold

$$\frac{1}{e}(\sqrt{p} - \sqrt{q})^2 < |d_k - d| < (\sqrt{p} - \sqrt{q})^2.$$
Problem

How good an approximation for $d$ is $d_k$?

Lemma

\[ |d_k - d| = \left(\frac{k}{e}\right)(\sqrt{p} - \sqrt{q})^2. \]

Since $k < e$, the following inequalities hold

\[(1/e)(\sqrt{p} - \sqrt{q})^2 < |d_k - d| < (\sqrt{p} - \sqrt{q})^2.\]

This shows that the smaller the value of $(\sqrt{p} - \sqrt{q})^2$, the quicker the attacker will arrive at a correct guess for the value of $d$. 
Let $n = pq$ be an $l$-bit RSA modulus. Let $e \geq 1$ and $\phi(n) \geq d$ satisfy $ed = 1 \mod \phi(n)$ and $e < 2^{\frac{n}{4} - 3}$. There is an algorithm that given $l/4$ least significant bits of $d$ computes all of $d$ in running time $O(en^c)$.

Boneh, Durfee, Frankel, An Attack on RSA Given a Small Fraction of the Private Key Bits, AsiaCrypt '98.