L.I.T.E.
Limited Information
Tetration Encryption

Darcy
@lvelieu

Ben
@puzzleduck

What's wrong with encryption?

- Based on computational complexity

- Everything encrypted today will one day be decryptable

- The richest can now buy decryption on demand

- SHA1 has fallen and SHA256 is in doubt

- Quantum computing is the answer we have explore

Can we fix it?

- An encryption scheme without dependency on computational power

- Dependant on fundamental laws of nature for security assurance

- Drop in replacement for RSA and SHA

- For reference there are 10^{80} atoms in the universe

- Tetration creates numbers in the order of $10^{15,151,335}$ using single digit inputs

Tetration What is it?

- Incrementing ++
  - $((3{+}{+}){+}{+}){+}{+}\ =\ 6$
- Addition + (Repeated increment)
  - $3+3\ =\ ((3{+}{+}){+}{+}){+}{+}\ =\ 6$
- Multiplication * (Repeated adds)
  - $3*3\ =\ 3+3+3\ =\ 9$
- Exponentiation ↑ / Power of ^ (Repeated multiply)
  - $3*3*3\ =\ 3^3\ =\ 3{\char`\^}3\ =\ 3{\uparrow}3\ =\ 27$
- Tetration / superexponentiation / hyperpower ↑↑ (Repeated exponent)
  - $3{\uparrow}(3{\uparrow}3)\ =\ 3{\uparrow}{\uparrow}3\ =\ 3\ {\char`\^}\ 3\ ...(27\ times)...\ {\char`\^}\ 3$
    - (result is > 5gb)

How does it work?

- Just like private key cryptography

- Private key: eg 58↑↑47 – Not capable of being stored in this universe

- Public key: subset of digits from Private key and a subset range of the result

- Encrypted data can utilise 100% of a channel to transmit a message of unknown length or values (Perfect forward security)

- Second order recursion (slow with standard computation)

Further Details

- Example C code in GitHub repository
- Worked example in text
- Exploration of concepts
- Presentation
- https://github.com/ivelieu/LITE
- https://ivelieu.github.io/LITE/

Questions ?
Comments ?
Suggestions ?

Darcy
@lvelieu

Ben
@puzzleduck