RSA Encryption Tools and Resources

Here is a curated list of **RSA encryption tools and resources**, categorized for different use cases, along with key features and security considerations:

---

1. Online RSA Encryption/Decryption Tools

Devglan RSA Tool
- Generate key pairs (supports 1024/2048/4096-bit keys) with X.509 public and PKCS8 private key formats.
- Encrypt/decrypt text using public or private keys, supports PKCS#1 and OAEP padding schemes.
- Includes OpenSSL command-line examples for key generation and file encryption.
Link

AnyCript RSA Tool
- Client-side encryption ensures data security within the user's device.
- Supports RSA modes like ECB/PKCS1 and ECB/OAEP, with key lengths from 512 to 4096 bits.
Link

dCode RSA Cipher
- Interactive tool for encryption/decryption with manual input of parameters (n, e, d).
- Includes helper utilities for calculating (n), modular inverse, and prime factorization.
Link

emn178's Online RSA Tool
- Encrypt messages using PKCS#1 or OAEP with SHA-1/SHA-256 hashing.
- Supports custom input/output encodings (UTF-8, Base64, Hex).
Link

---

2. Developer Resources

GeeksforGeeks Code Implementation
- Step-by-step C++, Java, and Python code examples for RSA key generation, encryption, and decryption.
- Demonstrates modular exponentiation and Euler's totient function calculations.
Link

OpenSSL Commands

Generate keys:

```
openssl genrsa -out private.pem 2048
openssl rsa -in private.pem -pubout -out public.pem
```

Encrypt/decrypt files:

```
openssl rsautl -encrypt -inkey public.pem -pubin -in data.txt -out data.enc
openssl rsautl -decrypt -inkey private.pem -in data.enc -out data.txt
```

---

3. Educational / CTF Tools

RsaCtfTool (GitHub)

- Designed for CTF challenges to attack weak RSA keys (e.g., small primes, Wiener's attack).
- Supports semiprime modulus factorization and kleptographic attacks.

Link

---

4. Key Security Considerations

- **Key Size**: Use **2048-bit or 4096-bit keys** for modern security. 1024-bit keys are deprecated.
- **Padding**: Use secure padding schemes like OAEP to prevent attacks.
- **Prime Generation**: Ensure primes (p, q) are large and random to avoid common modulus attacks.

---

5. Advanced Resources

RSA Factoring Challenges

- Historical contests (e.g., 2020 record of 829-bit factorization) highlight the difficulty of breaking RSA.

Okta's RSA Guide

- Explains real-world applications in digital signatures and TLS/SSL, with critiques on implementation flaws.

For a deeper dive into RSA's mathematical foundations (e.g., Euler's theorem, modular arithmetic) or enterprise solutions like RSA's Unified Identity Platform, refer to the cited sources.