Encryption Modes

Suppose we have a fixed symmetric encryption/decryption algorithm $E/D$ (presumably, with a key) operating on blocks of size $n$. We wish to encrypt a long message $P$. 
Electronic Codebook (ECB) Mode

Divide $P$ into blocks of size $n$: $p^1$, $p^2$, ....

Compute $c^i = E(p^i)$, $i = 1, 2, ....$

Send the sequence $c^1, c^2, ....$
Note that a plaintext block will encrypt the same way each time.

- Known plaintext: can start a dictionary
- Patterns in the plaintext sometimes evident in the ciphertext
- Can substitute ciphertext blocks

**Idea:** have each block affect the encryption of the following block: **Feedback** or **Chaining**
Cipher Block Chaining (CBC) Mode

\[ c^i = E(c^{i-1} \oplus p^i), \quad i = 1, 2, 3, \ldots \]

Decryption: \( p^i = D(c^i) \oplus c^{i-1} \)

\( c^0 = \text{IV} \) (initial vector)

© 2011 Clifford Bergman
CBC Mode

\[ \text{IV} \rightarrow p1 \rightarrow + \rightarrow E \rightarrow c1 \]
\[ \text{p2} \rightarrow + \rightarrow E \rightarrow c2 \]
\[ \text{p3} \rightarrow + \rightarrow E \rightarrow c3 \]
\[ \text{p4} \rightarrow + \rightarrow E \rightarrow c4 \]
\[ \ldots \]
To start this off, we choose an initial vector (IV) \( c^0 \). The IV can be a random block, a timestamp or a counter (a “nonce”). It can be sent in the clear.

It should be different for every message.
DEAR DAD,
FIRST DAY IN SCIENTIFIC FARMING WAS GREAT.
CAMPUS IS NICE.

LOVE,
BILLY

P.S. YOU’RE PLANTING THE CORN ALL WRONG.

Idea No. 2: Universities should teach practical subjects, like farming, engineering and home ec.
Sample picture encrypted in ECB mode

© 2011 Clifford Bergman
Sample picture encrypted in CBC mode

© 2011 Clifford Bergman
Output Feedback (OFB) Mode

Let $b^0$ be an IV. Create $b^1, b^2, \ldots$ by

$$b^i = E(b^{i-1}) \quad i = 1, 2, 3, \ldots$$

This can be done in advance of message generation.

Encryption: $c^i = p^i \oplus b^i$

Decryption: $p^i = c^i \oplus b^i$

The sequence of $b$’s acts as a one-time pad.

Different messages require either different keys or a different IV.
OFB Mode

\[ IV \rightarrow E \rightarrow b1 \rightarrow E \rightarrow b2 \rightarrow \ldots \]

\[ \begin{array}{c}
IV \\
E \\
b1 \\
E \\
b2 \\
+ \\
p1 \\
+ \\
c1 \\
+ \\
p2 \\
c2 \\
\ldots \\
\end{array} \]

© 2011 Clifford Bergman
Counter (CTR) mode

Very similar to OFB mode.

Choose an initial vector IV. Create $b^1, b^2, \ldots$ by

$$b^i = E(IV \parallel i) \quad i = 1, 2, 3, \ldots$$

Encryption: $c^i = p^i \oplus b^i$

Decryption: $p^i = c^i \oplus b^i$
Error Propagation

Suppose that, in transmission, one bit of $c^j$ is garbled. What effect will that have on the decrypted text?

- **ECB**: block $p^j$ will be completely corrupted.
- **OFB and CTR**: one bit of $p^j$ will be wrong.
- **CBC**: One block will be corrupted and one additional bit will be wrong.
Decryption in CBC mode
CBC mode can be used for message authentication. Any change to the plaintext will cause the last ciphertext block to be completely changed.
Security Considerations

Since OFB and CTR modes do not require the decryption function, they should not be used with public-key ciphers.

Don’t use ECB mode for large messages.

In OFB, it is possible for the key-stream to cycle prematurely. Very bad. This does not happen in CTR mode.
Collisions

Suppose that for some $i \neq j$, $c^i = c^j$.

In CBC: $p^i \oplus p^j = c^{i-1} \oplus c^{j-1}$

Note that for an $n$-bit block, expect a collision after approximately $2^{n/2}$ blocks.

Note that this can not happen in CTR mode.
Initial vector

Each message should use a new IV.

In OFB or CTR, repeated IV is a disaster
In CBC, get a collision

Instead of a simple counter, $i$, it is best to use something more random such as $E(i)$ or $H(i)$ for some hash function $H$. 