BILL, RECORD LECTURE!!!!

BILL RECORD LECTURE!!!

What Have We Learned From Classical Ciphers?

General Principles

General Principles

1. English, other languages, finance data, whatever all have patterns that can be used to help crack codes.

General Principles

1. English, other languages, finance data, whatever all have patterns that can be used to help crack codes.
2. **Kerchoffs's Principle** Eve knows the coding system that Alice and Bob are using. More generally, Alice and Bob should use a system whose security does not depend on Eve not knowing the system. **Caveat** It may be that when Alice and Bob **first** use a system Eve does not know what is is, which makes it more secure **temporarily**.

General Principles

1. English, other languages, finance data, whatever all have patterns that can be used to help crack codes.
2. **Kerchoffs's Principle** Eve knows the coding system that Alice and Bob are using. More generally, Alice and Bob should use a system whose security does not depend on Eve not knowing the system. **Caveat** It may be that when Alice and Bob **first** use a system Eve does not know what is is, which makes it more secure **temporarily**.
3. This course is mostly on the mathematics part of crypto. Some students have said **correctly**

What is we use Vig and then Matrix? Won't that be more secure?

General Principles

1. English, other languages, finance data, whatever all have patterns that can be used to help crack codes.
2. **Kerchoffs's Principle** Eve knows the coding system that Alice and Bob are using. More generally, Alice and Bob should use a system whose security does not depend on Eve not knowing the system. **Caveat** It may be that when Alice and Bob **first** use a system Eve does not know what is is, which makes it more secure **temporarily**.
3. This course is mostly on the mathematics part of crypto. Some students have said **correctly**

What is we use Vig and then Matrix? Won't that be more secure? The answer is **Yes** and people **Do do things like that**; however, this course won't deal with this. Also, see next point.

General Principles

1. English, other languages, finance data, whatever all have patterns that can be used to help crack codes.
2. **Kerchoffs's Principle** Eve knows the coding system that Alice and Bob are using. More generally, Alice and Bob should use a system whose security does not depend on Eve not knowing the system. **Caveat** It may be that when Alice and Bob **first** use a system Eve does not know what is is, which makes it more secure **temporarily**.
3. This course is mostly on the mathematics part of crypto. Some students have said **correctly**

What is we use Vig and then Matrix? Won't that be more secure? The answer is **Yes** and people **Do do things like that**; however, this course won't deal with this. Also, see next point.

4. If Eve does not know you are doing this double-encoding, then does add an extra layer of security. But by Kerchoffs's principle, Eve will know. But see next point.

General Principles

1. English, other languages, finance data, whatever all have patterns that can be used to help crack codes.
2. **Kerchoffs's Principle** Eve knows the coding system that Alice and Bob are using. More generally, Alice and Bob should use a system whose security does not depend on Eve not knowing the system. **Caveat** It may be that when Alice and Bob **first** use a system Eve does not know what is is, which makes it more secure **temporarily**.
3. This course is mostly on the mathematics part of crypto. Some students have said **correctly**

What is we use Vig and then Matrix? Won't that be more secure? The answer is **Yes** and people **Do do things like that**; however, this course won't deal with this. Also, see next point.

4. If Eve does not know you are doing this double-encoding, then does add an extra layer of security. But by Kerchoffs's principle, Eve will know. But see next point.
5. Double encoding will make Eve take **more time** to crack which is a mild win.

General Principles

1. English, other languages, finance data, whatever all have patterns that can be used to help crack codes.
2. **Kerchoffs's Principle** Eve knows the coding system that Alice and Bob are using. More generally, Alice and Bob should use a system whose security does not depend on Eve not knowing the system. **Caveat** It may be that when Alice and Bob **first** use a system Eve does not know what is is, which makes it more secure **temporarily**.
3. This course is mostly on the mathematics part of crypto. Some students have said **correctly**

What is we use Vig and then Matrix? Won't that be more secure? The answer is **Yes** and people **Do do things like that**; however, this course won't deal with this. Also, see next point.

4. If Eve does not know you are doing this double-encoding, then does add an extra layer of security. But by Kerchoffs's principle, Eve will know. But see next point.
5. Double encoding will make Eve take **more time** to crack which is a mild win.

The Shift, Affine, Quad Ciphers

The Shift, Affine, Quad Ciphers

1. Just saying **there are only 26 shifts** not enough to crack it.

The Shift, Affine, Quad Ciphers

1. Just saying **there are only 26 shifts** not enough to crack it. Also need IS-ENGLISH which is useful for both classical and modern crypto.

The Shift, Affine, Quad Ciphers

1. Just saying **there are only 26 shifts** not enough to crack it. Also need IS-ENGLISH which is useful for both classical and modern crypto.
2. **Key Space Principle** A large keyspace is a necc. condition to make a code uncrackable.

The Shift, Affine, Quad Ciphers

1. Just saying **there are only 26 shifts** not enough to crack it. Also need IS-ENGLISH which is useful for both classical and modern crypto.
2. **Key Space Principle** A large keyspace is a necc. condition to make a code uncrackable.
3. **Crackability** depends both on math and on technology. Affine was once hard to crack but is now easy to crack.

The Shift, Affine, Quad Ciphers

1. Just saying **there are only 26 shifts** not enough to crack it. Also need IS-ENGLISH which is useful for both classical and modern crypto.
2. **Key Space Principle** A large keyspace is a necc. condition to make a code uncrackable.
3. **Crackability** depends both on math and on technology. Affine was once hard to crack but is now easy to crack.
4. **Ease of Use** Its not enough for a system to be hard to crack. It must also be easy to use. That is why the quadratic cipher was never used, even 2000 years ago when it would have been harder to crack than affine.

The Shift, Affine, Quad Ciphers

1. Just saying **there are only 26 shifts** not enough to crack it. Also need IS-ENGLISH which is useful for both classical and modern crypto.
2. **Key Space Principle** A large keyspace is a necc. condition to make a code uncrackable.
3. **Crackability** depends both on math and on technology. Affine was once hard to crack but is now easy to crack.
4. **Ease of Use** Its not enough for a system to be hard to crack. It must also be easy to use. That is why the quadratic cipher was never used, even 2000 years ago when it would have been harder to crack than affine.
   Mini Project Actually code up and crack shift, affine, quadratic, and see what the gap is in the **IS-ENGLISH** program.

Gen Sub Cipher

Gen Sub Cipher

1. Just saying **freq Analysis** not enough to crack it.

Gen Sub Cipher

1. Just saying **freq Analysis** not enough to crack it.
   Mini Project Actually crack gen sub cipher.

Gen Sub Cipher

1. Just saying **freq Analysis** not enough to crack it.
   Mini Project Actually crack gen sub cipher.
2. Mathematics is not enough— some parameters have to be guessed.

Gen Sub Cipher

1. Just saying **freq Analysis** not enough to crack it.
   Mini Project Actually crack gen sub cipher.
2. Mathematics is not enough— some parameters have to be guessed.
3. Cryptography is **really used**. Any field that is **really used** has to have a combination of mathematics, empirical, and even ad-hoc guesswork.

Gen Sub Cipher

1. Just saying **freq Analysis** not enough to crack it.
   Mini Project Actually crack gen sub cipher.
2. Mathematics is not enough— some parameters have to be guessed.
3. Cryptography is **really used**. Any field that is **really used** has to have a combination of mathematics, empirical, and even ad-hoc guesswork.
   (if you do not like empirical and ad-hoc guesswork then take **CMSC 752: Ramsey Theory** with me in the Spring. Its not on Testudo yet but will be soon.)

Vig Ciphers

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.
   Mini Project Actually Crack Book-Vig.

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.
   Mini Project Actually Crack Book-Vig.
2. **Book-Vig** Do not use a book that Eve can guess. More generally, do not use anything that Eve can guess. Thats why (later) 1-time pad is so good, Eve literally cannot guess.

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.
   Mini Project Actually Crack Book-Vig.
2. **Book-Vig** Do not use a book that Eve can guess. More generally, do not use anything that Eve can guess. Thats why (later) 1-time pad is so good, Eve literally cannot guess.
3. **Vig** English Freq distributions hold even for (say) every 8th letter of a long normal text.

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.
   Mini Project Actually Crack Book-Vig.
2. **Book-Vig** Do not use a book that Eve can guess. More generally, do not use anything that Eve can guess. Thats why (later) 1-time pad is so good, Eve literally cannot guess.
3. **Vig** English Freq distributions hold even for (say) every 8th letter of a long normal text.
   Mini Project See how long a text you need so that this is true.

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.
   Mini Project Actually Crack Book-Vig.
2. **Book-Vig** Do not use a book that Eve can guess. More generally, do not use anything that Eve can guess. Thats why (later) 1-time pad is so good, Eve literally cannot guess.
3. **Vig** English Freq distributions hold even for (say) every 8th letter of a long normal text.
   Mini Project See how long a text you need so that this is true.
4. **Vig** Technology changes how we do things. To find length of key:

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.
   Mini Project Actually Crack Book-Vig.
2. **Book-Vig** Do not use a book that Eve can guess. More generally, do not use anything that Eve can guess. Thats why (later) 1-time pad is so good, Eve literally cannot guess.
3. **Vig** English Freq distributions hold even for (say) every 8th letter of a long normal text.
   Mini Project See how long a text you need so that this is true.
4. **Vig** Technology changes how we do things. To find length of key:
   Old Way Spotting a pattern that occurred many times used to be done by humans and required practice.

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.
   Mini Project Actually Crack Book-Vig.
2. **Book-Vig** Do not use a book that Eve can guess. More generally, do not use anything that Eve can guess. Thats why (later) 1-time pad is so good, Eve literally cannot guess.
3. **Vig** English Freq distributions hold even for (say) every 8th letter of a long normal text.
   Mini Project See how long a text you need so that this is true.
4. **Vig** Technology changes how we do things. To find length of key:
   Old Way Spotting a pattern that occurred many times used to be done by humans and required practice.
   New Way 1 Spotting a pattern that occurred many times by a program.

Vig Ciphers

1. **Book-Vig** uses that both Key and Text are English.
   Mini Project Actually Crack Book-Vig.
2. **Book-Vig** Do not use a book that Eve can guess. More generally, do not use anything that Eve can guess. Thats why (later) 1-time pad is so good, Eve literally cannot guess.
3. **Vig** English Freq distributions hold even for (say) every 8th letter of a long normal text.
   Mini Project See how long a text you need so that this is true.
4. **Vig** Technology changes how we do things. To find length of key:
   Old Way Spotting a pattern that occurred many times used to be done by humans and required practice.
   New Way 1 Spotting a pattern that occurred many times by a program.
   New Way 2 Try all lengths.

1-Time Pad, Keyword Shift, Linear Cong Gen

1-Time Pad, Keyword Shift, Linear Cong Gen

1. **1-Time Pad** uncrackable but needs truly random bits.

1-Time Pad, Keyword Shift, Linear Cong Gen

1. **1-Time Pad** uncrackable but needs truly random bits.
2. Want: from a small source of perhaps random bits, generated a longer string of psuedo-random bits.

1-Time Pad, Keyword Shift, Linear Cong Gen

1. **1-Time Pad** uncrackable but needs truly random bits.
2. Want: from a small source of perhaps random bits, generated a longer string of psuedo-random bits.
3. **Keyword Shift cipher** is one example of trying to generate a **random looking** sequence of bits. It also has a shorter key than gen-sub-cipher. It is no longer used since its just a gen-sub-cipher so crackable anyway.

1-Time Pad, Keyword Shift, Linear Cong Gen

1. **1-Time Pad** uncrackable but needs truly random bits.
2. Want: from a small source of perhaps random bits, generated a longer string of psuedo-random bits.
3. **Keyword Shift cipher** is one example of trying to generate a **random looking** sequence of bits. It also has a shorter key than gen-sub-cipher. It is no longer used since its just a gen-sub-cipher so crackable anyway.
   Mini Project Code up Keyword Shift cipher and see if its easier to crack then Gen Sub Cipher.

1-Time Pad, Keyword Shift, Linear Cong Gen

1. **1-Time Pad** uncrackable but needs truly random bits.
2. Want: from a small source of perhaps random bits, generated a longer string of psuedo-random bits.
3. **Keyword Shift cipher** is one example of trying to generate a **random looking** sequence of bits. It also has a shorter key than gen-sub-cipher. It is no longer used since its just a gen-sub-cipher so crackable anyway.
   Mini Project Code up Keyword Shift cipher and see if its easier to crack then Gen Sub Cipher.
4. **Linear Cong Gen** Used but crackable. Eve needs to know about topic, which she does.

1-Time Pad, Keyword Shift, Linear Cong Gen

1. **1-Time Pad** uncrackable but needs truly random bits.
2. Want: from a small source of perhaps random bits, generated a longer string of psuedo-random bits.
3. **Keyword Shift cipher** is one example of trying to generate a **random looking** sequence of bits. It also has a shorter key than gen-sub-cipher. It is no longer used since its just a gen-sub-cipher so crackable anyway.
   Mini Project Code up Keyword Shift cipher and see if its easier to crack then Gen Sub Cipher.
4. **Linear Cong Gen** Used but crackable. Eve needs to know about topic, which she does.
   Mini Project Code it up. Better with longer or shorter words to look for?

1-Time Pad, Keyword Shift, Linear Cong Gen

1. **1-Time Pad** uncrackable but needs truly random bits.
2. Want: from a small source of perhaps random bits, generated a longer string of psuedo-random bits.
3. **Keyword Shift cipher** is one example of trying to generate a **random looking** sequence of bits. It also has a shorter key than gen-sub-cipher. It is no longer used since its just a gen-sub-cipher so crackable anyway.
   Mini Project Code up Keyword Shift cipher and see if its easier to crack then Gen Sub Cipher.
4. **Linear Cong Gen** Used but crackable. Eve needs to know about topic, which she does.
   Mini Project Code it up. Better with longer or shorter words to look for?
5. **Mersenne Twister** Used but crackable. Eve needs to know about topic, which she does.

1-Time Pad, Keyword Shift, Linear Cong Gen

1. **1-Time Pad** uncrackable but needs truly random bits.
2. Want: from a small source of perhaps random bits, generated a longer string of psuedo-random bits.
3. **Keyword Shift cipher** is one example of trying to generate a **random looking** sequence of bits. It also has a shorter key than gen-sub-cipher. It is no longer used since its just a gen-sub-cipher so crackable anyway.
   Mini Project Code up Keyword Shift cipher and see if its easier to crack then Gen Sub Cipher.
4. **Linear Cong Gen** Used but crackable. Eve needs to know about topic, which she does.
   Mini Project Code it up. Better with longer or shorter words to look for?
5. **Mersenne Twister** Used but crackable. Eve needs to know about topic, which she does.
   Mini Project Code it up. Similar to LCG project.

Gen 2-Sub Cipher and Matrix

Gen 2-Sub Cipher and Matrix

1. **Gen 2-Sub Cipher** Timing is everything: There was never a time this was easy to use and hard to crack.

Gen 2-Sub Cipher and Matrix

1. **Gen 2-Sub Cipher** Timing is everything: There was never a time this was easy to use and hard to crack.
   Mini Project Write a program to crack Gen 2-sub cipher.

Gen 2-Sub Cipher and Matrix

1. **Gen 2-Sub Cipher** Timing is everything: There was never a time this was easy to use and hard to crack.
   Mini Project Write a program to crack Gen 2-sub cipher.
2. **Matrix Cipher** Cipher-text only might be uncrackable, but see next item.

Gen 2-Sub Cipher and Matrix

1. **Gen 2-Sub Cipher** Timing is everything: There was never a time this was easy to use and hard to crack.
   Mini Project Write a program to crack Gen 2-sub cipher.
2. **Matrix Cipher** Cipher-text only might be uncrackable, but see next item.
   Mini Project Write a program to crack matrix cipher-text only.

Gen 2-Sub Cipher and Matrix

1. **Gen 2-Sub Cipher** Timing is everything: There was never a time this was easy to use and hard to crack.
   Mini Project Write a program to crack Gen 2-sub cipher.
2. **Matrix Cipher** Cipher-text only might be uncrackable, but see next item.
   Mini Project Write a program to crack matrix cipher-text only.
3. **Matrix Cipher** If Eve has pairs of plaintext-cipher tex then she can **easily** crack the Matrix cipher. This makes us realize that we need to be careful on what we can assume Eve knows.

Gen 2-Sub Cipher and Matrix

1. **Gen 2-Sub Cipher** Timing is everything: There was never a time this was easy to use and hard to crack.
   Mini Project Write a program to crack Gen 2-sub cipher.
2. **Matrix Cipher** Cipher-text only might be uncrackable, but see next item.
   Mini Project Write a program to crack matrix cipher-text only.
3. **Matrix Cipher** If Eve has pairs of plaintext-cipher tex then she can **easily** crack the Matrix cipher. This makes us realize that we need to be careful on what we can assume Eve knows.
   Mini Project Write a program to crack matrix cipher given pairs. Requires you to write programs that deal with matrices mod 26. There are many Matrix Packages on the web and in Python, but they are for matrices over $\mathbb{Q}$ and cannot be adapted for mod 26.

Matrix Cipher Via Brute Force

Brute Force-matrix took $26^{O(n^2)}$.
Brute Force-Rows took $n26^{O(n)}$.

Matrix Cipher Via Brute Force

Brute Force-matrix took $26^{O(n^2)}$.
Brute Force-Rows took $n26^{O(n)}$.

1. Eve may have some trick you had not thought of.

Matrix Cipher Via Brute Force

Brute Force-matrix took $26^{O(n^2)}$.
Brute Force-Rows took $n26^{O(n)}$.

1. Eve may have some trick you had not thought of.
2. If Alice and Bob increase their value of n they can thwart Brute-Force-Rows, but

Matrix Cipher Via Brute Force

Brute Force-matrix took $26^{O(n^2)}$.
Brute Force-Rows took $n26^{O(n)}$.

1. Eve may have some trick you had not thought of.
2. If Alice and Bob increase their value of n they can thwart Brute-Force-Rows, but
   2.1 They need to know that they need to do that.

Matrix Cipher Via Brute Force

Brute Force-matrix took $26^{O(n^2)}$.
Brute Force-Rows took $n26^{O(n)}$.

1. Eve may have some trick you had not thought of.
2. If Alice and Bob increase their value of n they can thwart Brute-Force-Rows, but
   - 2.1 They need to know that they need to do that.
   - 2.2 This makes life harder for Alice and Bob which is still a mild win for Eve.

Auto Key Cipher

1. **Auto Key Cipher** is a great example of **Kerchoffs's Principle**:

Auto Key Cipher

1. **Auto Key Cipher** is a great example of **Kerchoffs's Principle**:
   1.1 If Eve does not know that the Auto-Key Cipher is being used then hard to crack.

Auto Key Cipher

1. **Auto Key Cipher** is a great example of **Kerchoffs's Principle**:
   - 1.1 If Eve does not know that the Auto-Key Cipher is being used then hard to crack.
   - 1.2 If Eve knows that the Auto-Key Cipher is being used then easy to crack.

Auto Key Cipher

1. **Auto Key Cipher** is a great example of **Kerchoffs's Principle**:
   1.1 If Eve does not know that the Auto-Key Cipher is being used then hard to crack.
   1.2 If Eve knows that the Auto-Key Cipher is being used then easy to crack.
2. **Auto Key Cipher** was never used since it was considered hard to use. So again, a cipher can't just be hard to crack, has to be easy to use.

Randomized Shift

Randomized Shift

1. **NY, NY Problem** We want that if NY appears many times in the text then it is coded different ways.

Randomized Shift

1. **NY, NY Problem** We want that if NY appears many times in the text then it is coded different ways.
2. If do this deterministically then need a long key.

Randomized Shift

1. **NY, NY Problem** We want that if NY appears many times in the text then it is coded different ways.
2. If do this deterministically then need a long key.
3. The only way to do this with a short key is to use randomization.

Randomized Shift

1. **NY, NY Problem** We want that if NY appears many times in the text then it is coded different ways.
2. If do this deterministically then need a long key.
3. The only way to do this with a short key is to use randomization.
4. **Randomized Shift** is a way to modify Shift to not have the NY,NY problem, though still crackable for other reasons, so not a serious cipher.

Randomized Shift

1. **NY, NY Problem** We want that if NY appears many times in the text then it is coded different ways.
2. If do this deterministically then need a long key.
3. The only way to do this with a short key is to use randomization.
4. **Randomized Shift** is a way to modify Shift to not have the NY,NY problem, though still crackable for other reasons, so not a serious cipher.
   Mini Project Code up Randomized-Shift, Affine, Matrix, and code up ways to crack.